

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису
УДК 519.216.3:007.5

До захисту допущено
В. о. завідувача кафедри ММСА

О.Л.Тимошук

«___» _____ 2018 р.

Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз
на тему: «Приховані Марківські ланцюги в задачах розпізнавання кризових
явищ»

Виконав:

студент II курсу, групи КА-71 мп

Пустовіт Дмитро Тарасович _____

Керівник:

Доцент кафедри ММСА, к.ф.-м.н., доцент

Каніовська Ірина Юріївна _____

Рецензент:

Доцент кафедри МАтаТЙ, к.ф.-м.н., доцент

Буценко Юрій Петрович _____

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів
без відповідних посилань

Студент _____

Київ
2018

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)

Спеціальність (спеціалізація) — 124 «Системний аналіз» («Системний аналіз і управління»)

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри ММСА

О. Л. Тимошук

«___» _____ 2018 р.

ЗАВДАННЯ

на магістерську дисертацію студенту Пустовіту Дмитру Тарасовичу

1. Тема дисертації: «Приховані Марківські ланцюги в задачах розпізнавання кризових явищ», науковий керівник дисертації Каніовська Ірина Юріївна, к.ф.-м.н., доцент, затверджені наказом по університету від «07» листопада 2018 р. № 4121-с

2. Термін подання студентом дисертації: _____

3. Об'єкт дослідження: кризове явище та його сутність.

4. Предмет дослідження: розпізнавання кризових явищ.

5. Перелік завдань, які потрібно розробити:

- 1) Огляд технічної літератури за темою роботи;
- 2) Дослідження актуальності обраної теми;
- 3) Побудова математичної моделі;
- 4) Збір вхідних даних;
- 5) Виконання обчислювальних експериментів;
- 6) Аналіз результатів моделювання і прогнозування;
- 7) Проведення аналізу ринкових можливостей запуску стартап-проекту;
- 8) Підготовка ілюстративного матеріалу;
- 9) Оформлення пояснювальної записки.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу:

- 1). Підготовлені дані
- 2). Побудовані моделі
- 3). Аналіз Результатів

7. Орієнтовний перелік публікацій:

- 1) Пустовіт Д. Т. Марківський підхід в задачі передбачення кризових явищ ринку праці в Україні // Міжнародний науковий журнал "Інтернаука". — 2017. — №10.

8. Дата видачі завдання: _____**Календарний план**

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
	Отримання завдання на дипломну роботу	07.09.2018 – 08.09.2018	виконано
	Огляд технічної літератури за темою	09.09.2018 – 10.10.2018	виконано
	Огляд сучасних бібліотек та програмних засобів для вирішення проблеми дослідження	11.10.2018 – 16.10.2018	виконано
	Побудова математичної моделі	17.10.2018 – 28.10.2018	виконано
	Збір та попередня обробка вхідних даних	29.10.2018 – 01.11.2018	виконано
	Виконання обчислювальних експериментів	02.11.2018 – 05.11.2018	виконано
	Аналіз результатів моделювання і прогнозування;	06.11.2018 – 15.11.2018	виконано
	Проведення аналізу ринкових можливостей запуску стартап-проекту	16.11.2018 – 18.11.2018	виконано
	Підготовка ілюстративного матеріалу	18.11.2018 – 20.11.2018	виконано
	Оформлення пояснювальної записки	21.11.2018 – 26.11.2018	виконано

Студент

Д.Т. Пустовіт

Науковий керівник дисертації

І.Ю. Каніовська

РЕФЕРАТ

Магістерська дисертація: 88 с., 23 рис., 36 табл., 1 додаток, 19 джерел.

Тема магістерської дисертації: «Приховані Марківські ланцюги в задачах розпізнавання кризових явищ».

Об'єктом дослідження є кризові явища, їх причини та сутність.

Предметом дослідження є розпізнавання кризових явищ.

Метою дослідження є аналіз кризи як явища в цілому, та економіці зокрема, дослідження існуючих підходів та методів раннього виявлення кризових ситуацій, побудова математичної моделі на базі прихованого Марківського ланцюга та створення програмного продукту – системи раннього сповіщення кризових явищ.

Результати та їх новизна: проведено порівняння існуючих методологій виявлення та прогнозування кризових явищ та існуючих Марківських моделей. Побудований прихований Марківський ланцюг, досліджено можливі обчислювальні проблеми та шляхи їх вирішення. Створено систему раннього сповіщення та прогнозування кризових явищ, обґрунтовано вибір параметрів для модифікованої моделі. Програмний продукт реалізовано за допомогою мови програмування Javascript. Надано рекомендації до подальших досліджень.

Теоретичною та методологічною основою дослідження є праці практикуючих експертів в сфері економіки, прийняття рішень та вчених в галузі математики.

Галузь використання: складні системи, що потребують раннього виявлення можливостей переходу в кризові стани.

ПРИХОВАНИЙ ЛАНЦЮГ МАРКОВА, СИСТЕМА РАНЬОГО СПОВІЩЕННЯ, РОЗПІЗНАВАННЯ КРИЗОВИХ ЯВИЩ, АЛГОРИТМ БАУМА-ВЕЛША, ЕМ АЛГОРИТМ.

ABSTRACT

Master's thesis: 88 p., 23 fig., 36 tabl., 2 appendix, 19 sources.

Topic of the research: «Hidden Markov models for complex crisis phenomena recognition».

Object of the research is crises, their causes and nature.

Subject of the research is crisis forecasting and detection.

The aim of the study is to analyze the economic crisis, explore the existing approaches and methods for early detection of crisis, build a mathematical model based on hidden Markov models and create an early warning system.

The results and their novelty: The comparison of existing methodologies for identifying and predicting crises and existing Markov models was carried out. There was also built hidden Markov chain, analyzed possible computing problems and their solutions. The early warning system of economic crisis in Ukraine was designed.

Software was developed with the usage of Javascript programming language. Recommendations for further research were given.

Theoretical and methodological basis of the study are works of practicing experts in the field of economics, decision making and scientists in the field of mathematics.

Uses: complex systems that require early detection of potential transition to critical states.

HIDDEN MARKOV MODELS, EARLY PREDICTING SYSTEM, EM ALGORITHM, BAUM-WELCH ALGORITHM, PREDICTING DOWNTURNS COMPLEX CRISIS PHENOMENA RECOGNITION

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 ПРИХОВАНІ МАРКІВСЬКІ ЛАНЦЮГИ У КОНТЕКСТІ НАЙГОСТРІШИХ СУСПІЛЬНИХ ПРОБЛЕМ 21 СТОЛІТТЯ	11
1.1 Актуальні задачі, що досліджуються з допомогою прихованих Марківських ланцюгів.	11
1.2 Огляд сучасних бібліотек та програмних засобів для роботи з прихованими Марківськими ланцюгами.	13
Висновки до першого розділу	17
РОЗДІЛ 2 КРИЗОВІ СИТУАЦІЇ ТА МЕТОДИ ЇХ ВИЯВЛЕННЯ	18
2.1 Поняття кризи. Суть та причини виникнення кризових явищ.	18
2.2 Передумови для створення систем раннього виявлення кризових явищ.	19
2.3 Огляд існуючих підходів вирішення задачі прогнозування кризових явищ.	20
Висновки до другого розділу	24
РОЗДІЛ 3 МАТЕМАТИЧНИЙ ОПИС МОДЕЛІ	25
3.1 Приховані ланцюги Маркова.....	25
3.2 Імовірність спостереження послідовності.....	30
3.3 Пошук найімовірнішої послідовності станів	33
3.4 Пошук параметрів моделі	35
3.5 Суміш Гаусівських компонент	43
3.6 Нормалізація моделі	52
Висновки до розділу	53
РОЗДІЛ 4 РЕЗУЛЬТАТИ РОБОТИ ПРОГРАМНОГО ПРОДУКТУ	55

4.1 Моніторинг соціальних мереж для визначення користувачів з ризиком депресії.....	55
Висновки до розділу	66
РОЗДІЛ 5 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ	68
5.1 Опис ідеї проекту	68
5.2 Технологічний аудит ідеї проекту.....	70
5.3 Аналіз ринкових можливостей запуску стартап-проекту.....	70
5.4 Розроблення ринкової стратегії проекту	77
5.5 Розроблення маркетингової програми стартап-проекту.....	81
Висновки до розділу	84
ВИСНОВКИ.....	85
ПЕРЕЛІК ПОСИЛАНЬ.....	86
ДОДАТОК А ЛІСТИНГ ПРОГРАМИ	89

ВСТУП

Історично, людина в процесі своєї діяльності прагне уникати хаосу, непередбачуваних ситуацій та нестабільності. Те ж саме стосується й поведінки складних систем. Будь-яка система прагне до стабільного стану. Стан рівноваги не передбачає існування зрушень, стрибків, розривів та хаосу. Але реалії завжди відрізняються від бажаного стану речей.

Останні 25 років показали, що не існує ефективного методу прогнозування кризових явищ, доказом цього стали хоча б 6 великих економічних криз з 1992 по 2008 рік. Відлуння економічної кризи 2008-2009 років ще довго тероризувало в тому числі й українську економіку. В Україні за останні кілька років крім виявлення кризових ситуацій в економіці з'явилася також нагальна потреба в системі раннього попередження для суспільних, політичних, соціальних та військових процесів.

Найпопулярнішим методом прогнозування кризових явищ безперечно є використання математичних моделей. Математичне моделювання дозволяє на основі існуючих даних (наприклад серії спостережень за об'єктом) передбачити майбутнє його параметрів, або поведінки в цілому.

Незважаючи на те, що на сьогоднішній день було запропоновано тисячі методологій та підходів до побудови математичної моделі, більшість з них виявляються неефективними в умовах ризиків, невизначеності інформації чи стрибкоподібних змін. Це спричинило появу недовіри цілого ряду дослідників до можливості існування єдиного математичного підходу виявлення кризових явищ, та обумовлює актуальність вибору даної теми.

Дещо розширивши погляд варто зазначити, що криза, в широкому розумінні, так само потрібна системі, що розвивається, як і стабільний стан. Кризу можна розглядати як переломний момент у розвитку системи, який дає простір для нового витка економічних змін. З такої точки зору особливо

корисним буде існування системи раннього сповіщення про кризу, яке дозволило б відвернути катастрофічні наслідки та зосередитись на тому, аби взяти від подібних ситуацій максимальну користь.

Тому метою дослідження є аналіз кризи як явища, огляд існуючих підходів та методів раннього виявлення кризових ситуацій, побудова математичної моделі на базі прихованого Марківського ланцюга та створення програмного продукту – системи раннього сповіщення кризових явищ.

Об'єкт дослідження: кризові явища, їх причини та сутність

Предмет дослідження: прогнозування та виявлення кризових явищ.

Методи дослідження: аналіз наукової та науково-популярної літератури й інтернет-джерел, метод математичного моделювання, контр приклад, аналіз і синтез математичних фактів, узагальнення і конкретизація. При побудові математичних моделей були використані: методи оптимізації, математичний аналіз та теорія ймовірностей.

На початку дослідження були поставлені такі завдання:

1. На основі аналізу наукової та науково-популярної літератури, інтернет-джерел, встановити суть поняття «криза», причини її виникнення та методи виявлення;
2. Здійснити огляд сучасних реалізацій прихованих марківських ланцюгів у прикладних бібліотеках;
3. Побудувати прихований Марківський ланцюг, вивести зручні формули для тренування та здійснення прогнозів;
4. Побудувати систему раннього попередження кризових явищ, здатну своєчасно виявляти небезпечні відхилення від норми досліджуваної системи.

Новизна даної роботи полягає в систематизованому аналізі проблеми виявлення та прогнозування кризових явищ, використанні сучасного математичного апарату для вирішення поставлених завдань.

Практичне значення одержаних результатів полягає в розробці системи раннього попередження та дієвої математичної моделі для прогнозування кризових явищ, з метою вдосконалення ефективності прийняття рішень у критичних ситуаціях.

Пропонована дипломна робота складається із вступу, п'яти розділів, висновків, списку використаних джерел (18 найменувань) та додатків.

РОЗДІЛ 1 ПРИХОВАНІ МАРКІВСЬКІ ЛАНЦЮГИ У КОНТЕКСТІ НАЙГОСТРІШИХ СУСПІЛЬНИХ ПРОБЛЕМ 21 СТОЛІТТЯ

1.1 Актуальні задачі, що досліджуються з допомогою прихованих Марківських ланцюгів.

Приховані ланцюги Маркова є насамперед дуже потужним методом, який використовують в задачах розпізнавання мови. Проте з часом, завдяки своїй простоті та гнучкості вони здобули широкого застосування в багатьох галузях. Тому цілком природньо, що існує цілий ряд актуальних задач, пов'язаний з неекономічними кризами, який успішно розв'язується з використанням цього інструменту.

Однією з найбільших проблем 21 століття є дослідження проблем міжнародних відносин, так криз у міжнародних відносинах, що призводять до військових конфліктів. Цій проблемі було присвячено ряд робіт Філіпа Щродта [9-11], в яких будувалися нелінійні моделі на базі прихованих марківський ланцюгів.

Зокрема у роботі [10] було проведено аналіз конфлікту на Близькому Сході за період з квітня 1979 року лютий 1997 року. В результаті чого було доведено що приховані ланцюги Маркова точніше класифікують міжнародні кризи ніж традиційні лінійні методи.

Успішним було й прогнозування конфліктів у Балканському конфлікті (війна в Югославії у період з січня 1991 по січень 1999)[9]. На основі заголовків сервісу новин Reuters тижні розмічались як такі, для яких імовірність конфлікту є високою, або ж низькою. Було протестовано схеми з позитивними (P) , негативними (N) вагами, а також без ваг (U). Відповідно U та P модель мали точність 25%, N модель – 50%, причому показувала точність 92% для короткострокових прогнозів.

Іншим прикладом став конфлікт у Лівії (1979-1997 роки) [11]. Тут автор запропонував дві моделі: LR та LRL приховані ланцюги. Відповідно вони мали б визначати чи переросте криза в війну чи ні. У підсумку було зроблено висновок, що LRL моделі дали точнішу класифікацію, ніж LR.

У роботі [5] ,було досліджено дуже болючу проблему для сучасної Європи – міграції населення. Дослідження проводилось на прикладі Німеччини. Було побудовано модель з двома станами – S_0 та S_1 – відповідно перебування мігранта в Німеччині та у своїй країні. Перехідні ймовірності були обчислені за формулою $P(S_{t+1} = i | S_t = j) = \frac{e^{\beta_{ij}x_{mt}}}{\sum_k e^{\beta_{ik}x_{mt}}}$, де коефіцієнти були розраховані на основі даних з німецької соціоекономічної панелі (GSOEP) – 12 тис. спостережень. Головним результатом, а за сумісництвом і головною несподіванкою дослідження стало те, що 80% депортованих мігрантів повертаються назад до Німеччини.

Моделі, побудовані на прихованих ланцюгах Маркова, знайшли своє застосування також і в аналізі ризиків при формуванні інвестиційного портфелю [2]. Зокрема у вказаній роботі було проведено аналіз за реальними даними 1970-2006 років. Для тренування моделі було використано ЕМ-алгоритм, а для прогнозування алгоритм Вітербі, та методи Прямого і Зворотнього Проходу, описані в другому розділі нашої роботи. Іншим застосуванням прихованих ланцюгів Маркова в економіці є аналіз фондового ринку[13,14]. В цих роботах в черговий раз доводиться твердження про доцільність Марківського припущення у роботі з фінансовими даними.

Цікаво, що Марківська модель може бути успішно застосована і до передбачення результатів виборів [4]. Зокрема в цій роботі будувалась матриця перехідних імовірностей для виборів в США 1988-2008 років. Проте не можна назвати побудовану модель надуспішною, адже за різних налаштувань вона не дозволяла передбачити більше ніж 3 з 6 результатів

(справедливості ради слід додати що й розрив між демократами й республіканцями лише на одних виборах перевищив 5%).

1.2 Огляд сучасних бібліотек та програмних засобів для роботи з прихованими Марківськими ланцюгами.

Серед спектру наукових бібліотек на мові програмування Python виділяється лише одна, для роботи з прихованими Марківськими ланцюгами. Це бібліотека з відкритим початковим кодом `hmmlearn`.

Вона поки не має стабільної мажорної версії, останній мінорний реліз (0.2.0) бібліотеки датований першим березня 2016 року. Остання стабільна версія на даний момент – 0.2.1.

Бібліотека зберігає API іншої відомої бібліотеки `scikit-learn`, та є по суті форком одного з її модулів. `Hmmlearn` може похвалитися 1.342 зірочками на гітхабі, де розміщено початковий код бібліотеки. Над нею працює команда з 23 розробників. `Hmmlearn` комерційно доступна під ліцензією BSD.

На офіційному сайті документації розміщено повний опис методів API а також приклади використання. Основними методами для роботи з бібліотекою є `fit` та `predict`. Перший Оцінює параметри моделі, а другий знаходить найімовірнішу послідовність станів для серії спостережень. Основною перевагою бібліотеки є те, що вона дозволяє використання суміші Гаусівських компонент.

Для `nodeJS` існує цілий ряд бібліотек призначених для роботи з прихованими Марківськими ланцюгами, проте жодній з них не вдалося здобути популярність. Вони пропонують лише обмежений список функцій (алгоритми `forward`, `backward`, `viterbi`), або мають документацію китаюською. Тому для цих бібліотек притаманні 10-20 зірочок на гітхабі, та останній комміт 3-4 роки тому.

Проте ситуація дещо змінилась з розвитком `tensorflow.js`. Зараз стрімко розвивається бібліотека `node-hidden-markov-model-tf`, побудована на основі `tensorflow`. Хоча вік бібліотеки всього три місяці, і вона має лише 4 зірочки на гітхабі, але є дуже амбіційним проектом. Її в середньому завантажують 400 разів на тиждень, а також вона написана з використанням сучасних технологій.

Над `node-hidden-markov-model-tf` наразі працює 3 розробники, а те що за нею спостерігають вже 30 учасників ком'юніті свідчить про її високий потенціал. На жаль документація на разі знаходиться в сирому стані, проте вже присутній приклад повного аналізу марківського ланцюга з використанням алгоритму Баума-Велша.

Найпопулярнішою бібліотекою для роботи з прихованими Марківськими ланцюгами в мові програмування R безперечно є `dermixS4`. Останньою задокументованою версією бібліотеки є 1.3-0.

Оцінка параметрів ланцюга в `dermixS4` здійснюється з допомогою `Newton-Raphson` оптимізованого варіанту ЕМ алгоритму. Також в тих випадках, коли використання ЕМ алгоритму є недоречним, в бібліотеці передбачена можливість прямої оптимізації з допомогою пакетів `Rsolnp` та `Rsolnp2`.

Бібліотека використовує спеціальний алгоритм нехтування відсутніми спостереженнями при розрахунку значень `forward` алгоритму.

В `dermixS4` реалізований стандартний для подібних засобів метод `fit`, а також спеціальний метод `mix` для специфікації моделі.

Приємним бонусом роботи з бібліотекою є ітераційне виведення результатів в процесі роботи алгоритму. До них належать правдоподібність та параметри моделі на кожній ітерації.

Пакет `dermixS4` також підтримує роботу з багатовимірними даними та Гаусівськими сумішами. Якщо цих можливостей з тієї чи іншої причини

виявиться недостатньо, `dermixS4` було спроектовано з можливістю розширення, та перевизначення всіх етапів роботи.

Результати огляду сучасних програмних засобів для роботи з прихованими Марківськими ланцюгами зведено до таблиці 1.1:

Таблиця 1.1 - Огляд сучасних програмних засобів для роботи з прихованими Марківськими ланцюгами

Бібліотека	Мова програмування	Функціонал	Ліцензія	Зірочок на Гітхабі	Завантажень на тиждень
<code>dermixS4</code>	R	Можливість використання Гаусівських сумішей			
		Можливість використовувати альтернативні методи оптимізації замість Алгоритму Баума-Велша			
		алгоритм нехтування відсутніми спостереженнями			

Продовження таблиці 1.1

Бібліотека	Мова програмування	Функціонал	Ліцензія	Зірочок на Гітхабі	Завантажень на тиждень
HiddenMarkov Model (0.0.5)	JavaScript	Підтримка лише дискретних станів	MIT	19	5
node-hidden-markov-model-tf (1.2.0)	JavaScript	Наявна підтримка Гаусівських сумішей	Apache-2.0	13	676
		Forward та Backward алгоритми			
		Можливість використання tensorflow бекенду			
hidden-markov-model (0.0.1)	JavaScript	Підтримка виключно Forward алгоритму	MIT	7	1
Hmmlearn (0.2.1)	Python	Наявна підтримка Гаусівських сумішей	BSD 3-Clause	1367	-

Висновки до першого розділу

Марківські ланцюги здобувають усе більшу популярність в якості основного інструментарію при вирішенні поставлених задач, ґрунтуючись на гіпотезі про незалежність майбутнього від всієї попередньої історії на етапі прогнозування.

Приховані марківські ланцюги активно використовуються в таких галузях як: моделювання військових конфліктів, передбачення криз в економіці, міграційних процесів та навіть використовуються для передбачення результатів президентських виборів.

В зв'язку з популярністю математичного апарату значного розвитку зазнали й програмні засоби імплементації алгоритмів прихованих Марківських ланцюгів, на популярних платформах NodeJs, R, Python.

РОЗДІЛ 2 КРИЗОВІ СИТУАЦІЇ ТА МЕТОДИ ЇХ ВИЯВЛЕННЯ

2.1 Поняття кризи. Суть та причини виникнення кризових явищ.

Взагалі визначення кризового явища, а особливо того, як можна відрізнити кризове явище від некризового - питання дуже суперечливе. На цей момент не існує єдиного критерію визначення кризових ситуацій – найчастіше таке поняття вживається задля словесних маніпуляцій, та слабо виражає реальний стан речей.

Як відзначають О. Н. Демчук і Т. А. Єфремова [1], криза виконує в динаміці руху систем три найважливіші функції:

1. різке ослаблення й усунення застарілих елементів пануючої системи, яка вже вичерпала свій потенціал;
2. розчищення дороги для затвердження спочатку слабких елементів нової системи, майбутнього циклу;
3. випробування на міцність і передача в спадщину тих елементів системи, які акумулюються й переходять у майбутнє.

Криза переживає декілька стадій у своїй динаміці:

1. Латентний період. - передумови кризи вже існують, але не мають яскравого зовнішнього вираження.
2. Період обвалу - у цей період набирають чинність, відкрито проявляються й вступають у боротьбу елементи наступної системи, що представляє майбутнє.
3. Період пом'якшення кризи - створення передумов для її подолання, переходу до фази депресії, що забезпечує тимчасову рівновагу між системою, що втратила чинність, і новою, що затвердилася.
4. Період прояву реальної можливості подолання кризи. На цьому етапі процес адаптації до кризи завершений, кризові процеси пом'якшилися завдяки локальній стабілізації підсистем, визначені основні програми виходу

із кризи. Відбуваються реструктуризація й відновлення основних підсистем, підсилюються позитивні й оптимістичні настрої, поліпшуються показники соціальної динаміки [1].

Ми використаємо результати кількісного аналізу кризових явищ, проведеного у роботі [18]. згідно з цими результатами кризою вважатимемо таку ситуацію, де значення досліджуваного індексу перевищує середнє значення більше ніж на одне стандартне відхилення

2.2 Передумови для створення систем раннього виявлення кризових явищ.

Останніми десятиліттями перед фінансовими аналітиками постав ряд нових задач, зумовлений інтеграцією та глобалізацією фінансових ринків у всьому світі. Цьому сприяло багато факторів. Перш за все це швидкий розвиток засобів зв'язку та переведення коштів з однієї валюти в іншу (відповідно з однієї держави чи навіть материка в інший). Саме тому сьогодні особи, що приймають рішення, мають все менше часу для того або вживати необхідних заходів при виникненні кризових явищ.

Слід перш за все зазначити що серед економістів практиків відсутня єдина думка щодо доцільності застосування математичного моделювання, як засобу здійснення прогнозів щодо ймовірності настання кризових явищ в фінансовій системі держави. Противники математичного підходу зазвичай посиляються на неточність прогнозів, а також на те, що надійність прогнозу знижується в період високої невизначеності стану економічної системи, тобто саме в тому стані, в якому прогнози є найціннішими.

На жаль наразі добре відомо, що фінансові ринки не здатні ефективно та своєчасно виявляти симптоми кризових явищ. Цьому було багато доказів лише за останні 25 років:

- атака на європейську монетарну систему 1992-1993 років;
- криза в Мексиці 1994 року;
- криза в Таїланді 1997 року;
- криза в Росії та південній Америці 1998 року;
- турецька банкова криза 2001 року;
- фінансова криза 2008 року.

У всіх вищезгаданих випадках брокери валютних бірж не вказували на можливі девальвації у ті місяці, що безпосередньо передували падінню валютних котирувань .

Відповідно до загальної економічної методології основними каналами передачі кризових явищ є наступні 3 ланки [15]:

- макроекономічна - виникнення загальних зрушень в одному регіоні може стати причиною одночасної появи кризи в різних країнах. Прикладом цього може бути подорожчання долара, або ж зниження цін на експортні товари;
- торгова - передача кризи торговим партнерам пояснюється через зниження конкурентоспроможності країни що вражена кризою;
- фінансова - виникнення кризи в одній або групі країн стимулює інвесторів продавати активи, які мають високий рівень кореляції з активами країни, що вражена кризою.

2.3 Огляд існуючих підходів вирішення задачі прогнозування кризових явищ.

У світовій практиці існує кілька основних до підходів розв'язання проблем кризового моделювання.

Перш за все варто відзначити роботи яких увага зосереджується на створенні та впровадженні низки випереджаючих індикаторів для банківських, фінансових та валютних криз.

Прикладом дослідження впливу індикаторів може слугувати [15]. В цій роботі було запропоновано проводити аналіз та прогнозування економічних індикаторів на базі декомпозиції часового ряду, аналітичного вирівнювання тренду, Фур'є аналізу, адаптивних методів прогнозування, моделях авторегресії. Окрему увагу було виділено моделі з інтегрованим ковзним середнім.

Найбільш поширеною загальною методологією моделювання кризових явищ в економіці є передусім Методологія економетричного моделювання - вона базується на застосування логіт- і пробіт- моделей.

Основними перевагами цієї методології є можливість оцінки ймовірності настання кризи та можливість одночасного аналізу декількох змінних. Проте вона має ще й ряд обмежень, через які так і не здобула всезагального використання. До цих обмежень належать у першу чергу відсутність можливості ранжування економічних індикаторів (виходячи з їхньої здатності прогнозувати настання кризи), нелінійна природа даних моделей - що багатьох випадках значно ускладнює оцінку ефективності індикатора та нестабільність у пошуках неефективних індикаторів.

Іншою популярною методологією є Сигнальний підхід (методологія непараметричних оцінок). Це методологія вважається найкращою альтернативою до методології економетричного моделювання. Їй не притаманні недоліки попереднього підходу, проте ефективних результатів застосування цієї методології так і не було виявлено, тому можна зробити висновок, що вона вимагає глибшого вивчення та дослідження нових моделей, які могли б дати їй переваги, спів вимірні з перевагами методології економетричного моделювання.

Зокрема у роботі [17] наголошується на тому що використання для прогнозів різних індексів є непридатним для оцінки можливості критичних події кризових явищ. Показником цьому сенсі є той факт, що при аналізі української економіки за допомогою регресійних моделей рекомендується виключати дані за 2009 рік, як такі, що не характерні для економіки України.

Іншим популярним підходом є імітаційне моделювання на базі концепції системної динаміки Дж. Форрестера. Наприклад у роботі [17] Було здійснено побудову імітаційної моделі розвитку кризових процесів у реальному секторі економіки України, виявлено причинно-наслідкові зв'язки та проведено аналіз різних сценаріїв можливої протидії кризовим явищам. Основними змінними блоків імітаційної моделі стали: обсяг випуску продукції, ВВП, експорт продукції, імпорт продукції, ємність внутрішнього ринку, кредити, інвестиції, дисбаланси, диспропорції, кредитний пузир та індекс промислового виробництва. В ході даного дослідження було розглянуто моделі виникнення криз у: сільському господарстві, виробництві продуктів нафтопереробного комплексу, машинобудуванні, металургійному виробництві. Основне припущення про залежність приросту обсягів випуску від інвестицій та кредитів в більшості випадків не підтвердилась, що пояснюється екстенсивним розширення виробництва на тимчасово невикористаних потужностях.

При цьому імітаційна частина моделі отримала високий ступінь адекватності (за критерієм середньовідсоткової помилки 5-10%). Слід зазначити, що аналіз зворотних зв'язків впливу змінних на динаміку показників функціонування галузей не було виявлено, хоча для всіх них індекс промислового виробництва негативно залежить від будь-якої диспропорції, не зважаючи на те що він не є головним індикатором настання кризи.

Популярності також набуває метод, заснований на побудові експертних систем, що базуються на методі аналізу ієрархій Сааті. Окреме місце посідають методи дослідження розвитку економічної динаміки з використанням виробничих функцій.

Тобто виходячи з останніх досліджень можна зробити висновок, що дослідники зараз в основному активно намагаються переглядати фактори, які сприяють фінансовим кризам. Іншим ключовим завданням є прагнення розробити систему раннього сповіщення та попередження, яка б змогла сигналізувати про можливі труднощі на валютних ринках.

Найчастіше систему раннього попередження створюють таким чином, щоб вона могла перевести інформацію про можливе настання кризи в певний сукупний показник вразливості.

Такі системи були б ефективними в якості засобу передчасного виявлення великих зрушень та стрибків, та надавали б допомогу центральному банком фінансовим інститутам та приватним компаніям уникнути великих втрат. Давали б необхідний час задля організації необхідних заходів - реакції на кризу, якомога швидше.

Однією з головних причин виникнення проблем при застосуванні математичного прогнозування до кризових явищ економічні та фінансові системах є те, що якщо вихідна статистична інформація має високий ступінь недостовірності то результат моделювання також матиме значний ступінь невизначеності.

Для розв'язання проблеми передбачення економічних криз приховані ланцюги Маркова було вперше використано у 2002 році для побудови системи раннього сповіщення для країн південно-східної Азії [1]. Зокрема було побудовано моделі для Індонезії Малайзії Філіппін та Тайланду. Розроблений у даному контексті підхід мав би забезпечити кращу роботу ніж класичні методології непараметричних оцінок та економетричного

модельовання, зокрема була поставлена мета уникнути можливих проблем з класифікацією індикаторів. Побудовані моделі були Марковськими моделями з перемикачем, виходом яких була авторегресія. Отриманий ланцюг Маркова мав два стани (криза та не криза). Відповідно другою частиною моделі була авторегресія деяких параметрів (які були унікальними для кожної країни), та в загальному випадку записувалася так:

$$y_t = \alpha_{st} + x_t \beta_{st} + \sigma_{st} \varepsilon_t$$

Висновки до другого розділу

Події двох останніх десятиліть показали, що людство все ще не здатне в повній мірі передбачати та зустрічати кризові явища. Окреме місце тут займає потреба в існуванні системи раннього попередження, яка могла б бути успішно використана на всіх можливих рівнях: від глобальної економічної кризи, до виявлення кризових явищ на окремо взятому підприємстві. Існуючі немарківські моделі дедалі частіше показують свою неспроможність в задачах прогнозування кризових явищ в економіці, соціальній, демографічній сферах.

Однією з головних причин виникнення проблем при застосуванні математичного модельовання до прогнозування кризових явищ є те, що якщо вихідна статистична інформація має високий ступінь недостовірності то й результат модельовання також матиме значний ступінь невизначеності.

РОЗДІЛ 3 МАТЕМАТИЧНИЙ ОПИС МОДЕЛІ

3.1 Приховані ланцюги Маркова

Завдяки своїй простоті ланцюги Маркова набули чималої популярності в аналізі даних, особливо в роботі з послідовностями. Проте примітивна модель, яка була описана вище має також ряд обмежень, через що є неприпустимою для застосування в реальних задачах.

Яким чином ми можемо побудувати модель для послідовності станів, у тому випадку, якщо ми безпосередньо не спостерігаємо за цими станами, але можемо спостерігати за певною імовірнісною функцією над ними.

Як не дивно, проте вперше з розв'язанням цієї проблеми зіткнулись дослідники розпізнавання мови. Наприклад в задачах побудови сценаріїв розпізнавання частин мови звукова послідовність що спостерігається, не є послідовністю станів - частин мови, але її можна об'єднати в слова, які вже можна представити як певні ймовірнісні функції над частинами мови.

Наведемо простий класичний приклад необхідності створення спеціальної математичної теорії для таких випадків, під назвою «Кліматологія Морозива» (Ice cream climatology) [6]:

Кліматологи майбутнього в 2799 році досліджують проблеми глобального потепління. Для цього вони хочуть відновити дані про стан погоди у місті Балтимор у червні 2017 року. на жаль офіційних погодних даних з 2017 року у майбутньому не збереглося. Проте вони знайшли щоденник Джейсона Ейнера, в якому він щодня записував, скільки морозива він з'їдав за день. Цей щоденник є єдиним доступним для них джерелом у цій місцевості за даний період, відповідно за цими записами і необхідно відновити втрачені дані про стан погоди.

В такій ситуації ми не можемо стверджувати, що спостерігаємо за послідовністю станів (погодою в конкретні дні), проте вочевидь дані що збереглися в щоденнику є функціями від погоди (тобто станів ланцюга Маркова). Спостережувана послідовність виражатиме кількість вжитого морозива, і буде цілочисельною, наприклад $\vec{x} = \{1, 2, 2, 3, 4, 2, 1, 1, 2\}$. При цьому нам цікаві наступні стани ланцюга маркова $S = \{H, C\}$, де H – тепла погода, а C – холодна погода.

Схематично такий марківський представлено на рисунку 3.1:

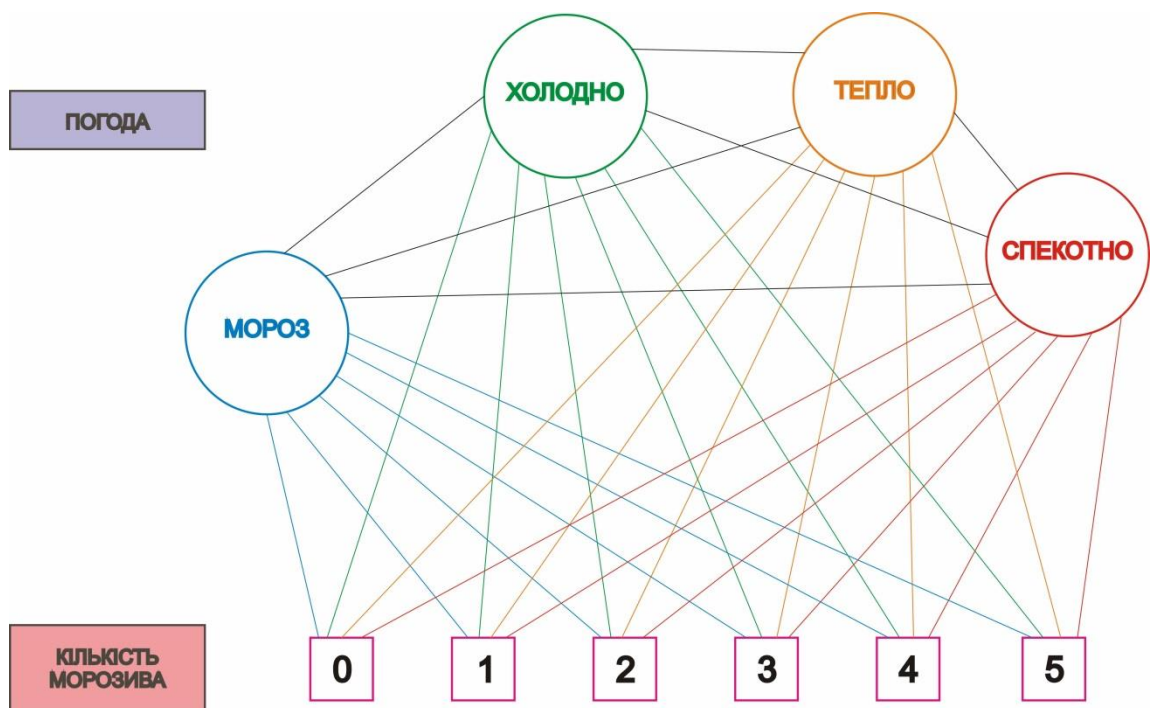


Рисунок 3.1 – Кліматологія морозива

Отже, як бачимо виникає необхідність розділити інформацію, що ми отримуємо – спостережувану послідовність та реальні стани ланцюга Маркова, які в цьому разі будуть прихованими. Таку модель називають прихованим ланцюгом Маркова.

Формально прихований марківський ланцюг є розширенням ланцюга Маркова, в якому вхідна послідовність символів відрізняється від

послідовності станів. Відповідно хоча ми формально й описуємо послідовність станів з певного алфавіту станів, проте в даному разі вже не спостерігаємо за цими значеннями.

Основною характеристикою прихованого ланцюга Маркова є матриця перехідних ймовірностей A , що представляє собою імовірності переходів між станами ланцюга:

$$A_j^i = \mathcal{P}\{s_{t+1} = i \mid s_t = j\}$$

Окрім того тепер присутня вихідна послідовність що безпосередньо спостерігається. Вихідні дані отримані з деякого вихідного алфавіту V . Спостереження x в момент часу t вважатимемо згенерованим деякою ймовірнісною функцією $b_j(x_t)$, яка буде пов'язана з станом j такою ймовірністю:

$$b_j(x_t) = \mathcal{P}\{x_t, s_t = j\}$$

В прихованому ланцюзі Маркова ми нічого не знаємо про згенеровані дані та їх зв'язок з моделлю. Зокрема для нас невідомими є матриця перехідних ймовірностей кількість станів, належність спостереження до певного стану, невідомою є навіть імовірність належності спостереження до конкретного стану. Зрештою така постановка є досить логічною, адже при роботі з реальними даними ми не можемо стверджувати що спостерігаємо кризове явище чи ні, навіть у разі якщо маємо доступ до всіх показників системи одночасно.

Тож замість того аби прив'язувати кожен стан ланцюга Маркова до певного спостереження, ми прив'яжемо кожен стан прихованого ланцюга Маркова до певної ймовірнісної функції. Такий підхід виявляється особливо

корисним у тих випадках коли аналізовані дані, особливо фінансові, отримані в результаті певного невідомого стохастичного процесу.

Методи, що побудовані на прихованих ланцюгах Маркова значно потужніші, за ті що побудовані на звичайних марківських ланцюгах. Приховані ланцюги Маркова здобули широкого застосування в багатьох галузях. Зокрема вони використовуються в задачах розпізнавання сигналів, людської мови, аналізу погодних даних, та фінансових часових рядів.

Зважаючи на таку гарну пристосованість прихованих ланцюгів Маркова до аналізу часових рядів, постає гостра необхідність відшукування швидкого та ефективного алгоритму тренування моделі та здійснення прогнозів.

Також для моделювання ймовірностей вихідними спостереженнями нам необхідно буде зробити припущення Маркова про незалежність і для вихідних змінних:

$$\mathcal{P}\{x_t = v_k \mid s_t = j\} = \mathcal{P}\{x_t = v_k \mid \vec{x}, \vec{s}\}$$

Тепер можемо визначити матрицю переходів прихованих станів за аналогією до матриці матриця перехідних імовірностей (A):

$$B_{x_t}^j = \{b_j(x_t)\} = \mathcal{P}\{x_t \mid s_t = j\}$$

Зауважимо що в даному випадку $B_{x_t}^j$ вважається матрицею, проте в формулах (3.8-3.12) буде запропоновано більш гнучкий підхід, в якому операції здійснюватимуться не над матрицею $B_{x_t}^j$, а безпосередньо над функціями $\{b_j(x_t)\}$.

Отже, формально прихований ланцюг Маркова можна представити у вигляді наступної структури: $\{S, V, \Pi, A, B\}$, де:

а) $S = \{1, \dots, N\} = \{s_1, \dots, s_N\}$ – це множина станів. Надалі стан прихованого ланцюга у момент t позначатимемо: s_t . А стан під номером i як i .

б) $V = \{v_1, \dots, v_M\}$ – це вихідний алфавіт. В дискретному випадку M – це кількість можливих значень які ми можемо спостерігати, в неперервному $V \subset \mathbb{R}$, а M матиме інший зміст.

в) $\Pi = \{\pi_i\}_{i \in S}$ – початковий розподіл, тобто $\pi_i = \mathcal{P}\{s_1 = i\}$

г) $A = \{A_j^i\}_{i,j \in S}$ – матриця ймовірнісних переходів (матриця перехідних імовірностей), тобто:

$$A_j^i = \mathcal{P}\{s_{t+1} = i | s_t = j\}, \quad 1 \leq i, j \leq N$$

д) $B_{x_t}^j = \{b_j(x_t)\}$ – розподіл ймовірності спостережень, тобто:

$$B_{x_t}^j = \mathcal{P}\{x_t = k, s_t = j\}, \quad k \in V$$

Існує три основних питання щодо прихованого ланцюга Маркова, які необхідно вирішити для подальшої роботи з моделлю:

1. За заданою моделлю $\mu = \{S, K, \Pi, A, B\}$, та поточними вхідними даними (послідовністю $\vec{x} \in V^T$) з'ясувати яка імовірність того що вказана послідовність спостережень $\vec{x} \in V^T$ відповідає моделі $\mu = \{S, K, \Pi, A, B\}$, тобто знайти $\mathcal{P}\{\vec{x} | \mu\}$.
2. За заданою моделлю $\mu = \{S, K, \Pi, A, B\}$, та поточними вхідними даними (послідовністю $\vec{x} \in V^T$) знайти таку послідовність станів $\vec{s} \in S^T$, яка б найкраще відповідала моделі та даним. Іншими словами, слід знайти: $\vec{s}^* = \operatorname{argmax}_{\vec{s}} \mathcal{P}\{s | \vec{x}, \mu\}$

3. За заданими вхідними даними (послідовністю $\vec{x} \in V^T$), знайти найкращі параметри моделі $\mu = \{S, K, \Pi, A, B\}$, іншими словами знайти таку модель, для якої $\mathcal{P}\{\vec{x} | \mu\} \rightarrow \max$.

3.2 Імовірність спостереження послідовності

Як вже було сказано раніше, у прихованому ланцюзі Маркова дані, які ми спостерігаємо можна вважати результатом роботи певного процесу, а послідовність спостережень (\vec{x}) можна вважати функцією над серією станів (\vec{s}) ланцюга Маркова з матрицею перехідних імовірностей A . В кожен момент часу ми спостерігаємо за функцією $s_t \rightarrow x_t$, тож для того аби знайти імовірність того що послідовність спостережень \vec{x} була отримана внаслідок роботи процесу, нам необхідно підрахувати функцію правдоподібності у вигляді суми правдоподібностей даних \vec{x} при всіх можливих послідовностях станів \vec{s} .

$$\mathcal{P}\{\vec{x}, A, B\} = \sum_s \mathcal{P}\{\vec{x}, \vec{s}, A, B\} = \sum_{\vec{s}} \mathcal{P}\{\vec{x} | \vec{s}, A, B\} * \mathcal{P}\{\vec{s}, A, B\}$$

Поки що в перетвореннях ми використали лише формулу умовної ймовірності, тож ця формула дійсна для будь-якого розподілу, проте ми можемо суттєво спростити подальші міркування та отримати зручні формули для обчислень, якщо застосуємо умову Маркова:

$$\mathcal{P}\{\vec{x}, A, B\} = \sum_{\vec{s}} \mathcal{P}\{\vec{x} | \vec{s}, A, B\} * \mathcal{P}\{\vec{s}, A, B\} =$$

$$= \sum_{\vec{s}} \prod_{t=1}^T \mathcal{P}\{s_t | x_t, B\} \prod_{t=1}^T \mathcal{P}\{s_t | s_{t-1}, A\} = \sum_{\vec{s}} \prod_{t=1}^T B_{X_t}^{s_t} \prod_{t=1}^T A_{s_t}^{s_{t-1}} \quad (3.1)$$

Отримали простий вираз, що може бути обчислений з наших початкових даних та ззовні нагадує аналогічний результат для простих ланцюгів Маркова. ($\mathcal{P}\{\vec{s}\} = \prod_{t=1}^T A_{s_t}^{s_{t-1}}$)

Для отримання формули (3.1) були послідовно використані незалежність елементів вихідної послідовності, умова Маркова, та стаціонарність випадкового процесу.

Єдиним недоліком даного підходу є те, що нам необхідно обчислювати суму по всіх можливих станах s . Враховуючи що кожен елемент послідовності s_t може набувати $|S|$ можливих значень, обчислення цієї суми матиме складність $O(|S|^T)$.

На щастя динамічне програмування в даному випадку надає швидкий спосіб обчислення такої суми, з допомогою алгоритму Прямого Проходу (англ. Forward Procedure)

Визначимо додаткову змінну, що виражатиме загальну ймовірність отримати ланцюжок спостережень за умов, що $s_t = i$ (тобто ми перебуваємо в стані i в момент часу t)

$$\alpha_i(t) = \mathcal{P}\{x_1, x_2, \dots, x_T, s_t = i, A, B\}$$

Тепер шукана ймовірність може бути представлена у такому вигляді:

$$\mathcal{P}\{\vec{x}, A, B\} = \sum_{i=1}^{|S|} \mathcal{P}\{x_1, x_2, \dots, x_T, s_t = i, A, B\} = \sum_{i=1}^{|S|} \alpha_i(T)$$

А власне сам пошук $\alpha_i(t)$ здійснюється з допомогою алгоритму Прямого Проходу.

Ініціалізація:

$$\alpha_i(0) = \pi_i, \quad i = 1..|S|$$

Рекурсія:

$$\alpha_j(t) = \sum_{i=1}^{|S|} \alpha_i(t-1) * A_j^i * B_{x_t}^i, \quad j = 1..|S|, \quad t = 1..T$$

Повністю аналогічно можна отримати й імовірності $\beta_i(t) = \mathcal{P}\{x_T, \dots, x_1, s_t = i, A, B\}$. Пошук $\beta_i(t)$ зветься Зворотнім Проходом (Backward Procedure).

Ініціалізація:

$$\beta_i(T) = 1, \quad i = 1..|S|$$

Рекурсія:

$$\beta_j(t) = \sum_{i=1}^{|S|} \beta_i(t+1) * A_j^i * B_{x_t}^i, \quad j = 1..|S|, \quad t = 1..T$$

Процедура виконується рекурсивно, з часом пропорційним до $T * |S|$.

3.3 Пошук найімовірнішої послідовності станів

Наступне важливе питання яке виникає при побудові прихованого ланцюга Маркова – яка послідовність станів \vec{s} , буде найімовірніше отримана для даної моделі та послідовності спостережень. Слід особливо зауважити, що ми мусимо зафіксувати кількість станів, адже наша ціль знайти таку послідовність $\vec{s} \in S^T$, за умови що чи спостерігаємо за $\vec{x} \in V^T$:

$$\operatorname{argmax}_{\vec{s}} \mathcal{P}\{\vec{z}|\vec{x}, A, B\} = \operatorname{argmax}_{\vec{s}} \frac{\mathcal{P}\{\vec{x}|\vec{s}, A, B\}}{\mathcal{P}\{\vec{x}, A, B\}} = \operatorname{argmax}_{\vec{s}} \mathcal{P}\{\vec{s}, \vec{x}, A, B\}$$

Остання рівність випливає з того, що $\mathcal{P}\{\vec{x}, A, B\} = \sum_{\vec{s}} \mathcal{P}\{\vec{x}, \vec{s}, A, B\}$ не залежить від \vec{s} безпосередньо.

Для пошуку $\operatorname{argmax}_{\vec{s}} \mathcal{P}\{\vec{s}, \vec{x}, A, B\}$ теоретично можна перебрати всі можливі послідовності \vec{s} , та вибрати оптимальну, проте складність такого алгоритму також дорівнюватиме $O(|S|^T)$. На цьому можна було б зупинитись, але як і для Прямого/Зворотнього Проходу, в даному випадку можна скористатись готовим рішенням з динамічного програмування. Спершу, як і у випадку з Прямим та Зворотнім Проходом визначимо додаткові змінні:

$$\delta_j(t) = \max_{s_1 \dots s_{t-1}} \mathcal{P}\{\vec{z}, s_1 \dots s_{t-1}, s_t = i | A, B\}$$

Ця змінна представляє собою імовірність того що спостерігається послідовність \vec{z} , найбільш імовірна послідовність (в динамічному програмуванні може називатись шляхом) станів завершується станом i в момент часу t , за умови що матриця перехідних імовірностей A , і матриця ймовірностей спостережень B .

$$\psi_j(t) = \underset{1 \leq i \leq N}{argmax} \{ \delta_j(t-1) A_j^i \}$$

Дана змінна зберігатиме вершину, що наразі найкраще підходить для включення в найімовірніший маршрут.

Ініціалізація:

$$\begin{aligned} \delta_i(0) &= \pi_i * B_{x_1}^i, & i &= 1..|S| \\ \psi_i(0) &= null, & i &= 1..|S| \end{aligned}$$

Індукція:

$$\begin{aligned} \delta_i(t) &= B_{x_{t-1}}^i * \underset{1 \leq j \leq N}{max} \{ \delta_j(t-1) A_i^j \}, & i &= 1..|S| \\ \psi_i(t) &= \underset{1 \leq j \leq N}{argmax} \{ \delta_j(t-1) A_i^j \}, & i &= 1..|S| \end{aligned}$$

Таким чином збільшуючи на кожному кроці індукції $t = t + 1$, ми знайдемо послідовність станів \vec{s} , буде найімовірніше отримана для даної моделі та послідовності спостережень:

$$\begin{aligned} P^* &= \underset{1 \leq j \leq N}{max} \{ \delta_j(T) \} \\ s_T^* &= \underset{1 \leq j \leq N}{argmax} \{ \delta_j(T) \} \\ s_t^* &= \psi_i(s_{t-1}^*) \end{aligned}$$

Даний алгоритм носить ім'я американського дослідника-інженера Андрія Джеймса Вітербі (англ. Viterbi Algorithm). Він є аналогічним Прямому Проходу, але замість обчислення сумарної імовірності, в даному випадку мусимо обрати з серії спостережень максимальну ймовірність.

Іншими словами, алгоритм Вітербі можна отримати заміною сумування за попередніми станами на максимізацію.

3.4 Пошук параметрів моделі

Останнім питанням що залишають приховані ланцюги Маркова є власне обчислення параметрів моделі. Це найважче питання, адже для того аби знайти найімовірнішу послідовність станів або перевірка відповідності вихідних даних до побудованої моделі неможлива, якщо невідомі параметри $\{S, K, P, A, B\}$.

ЕМ алгоритм (англ. Expectation Maximisation algorithm) – алгоритм, призначений для пошуку локального максимуму функції правдоподібності в моделі з прихованими змінними. Він проводиться ітераційно з деякого початкового значення, використовуючи наступні два кроки:

Е-крок (англ. Expectation step):

Знаходимо значення прихованих змінних, виходячи з апостеріорних розподілів, що представлені в даних.

М-крок (англ. Maximisation step):

Максимізуємо правдоподібність, припускаючи що ми спостерігаємо і за прихованими змінними.

В цьому параграфі приймемо таку нотацію:

$Y = \{y_i\}$ – спостережувані змінні

$X = \{x_i\}$ – приховані змінні

$\theta = \{\theta_i\}$ – шукані параметри

Зазвичай при дослідженні максимального значення зручнішою для дослідження виявляється не просто функція правдоподібності, а її логарифм.

$$\begin{aligned}
L(\theta) = \log(\mathcal{P}\{Y|\theta\}) &= \log\left(\sum_x \mathcal{P}\{X, Y|\theta\}\right) = \log\left(\sum_x q(x) * \frac{\mathcal{P}\{X, Y|\theta\}}{q(x)}\right) \geq \\
&\geq \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X, Y|\theta\}}{q(x)}\right)
\end{aligned} \tag{3.2}$$

Тут ми ввели додаткову функцію $q(x)$, $q(x) > 0$, для $\forall x$, після чого була використана відома нерівність:

Нерівність Єнсена (англ. Jensen's inequality)

Для дійсної опуклої (ввігнутої) функції φ , чисел $\{x_i\}_{i=1}^n$, та невід'ємних чисел α_i , таких що $\sum_i \alpha_i = 1$ має місце нерівність:

$$\begin{aligned}
\varphi\left(\sum_i \alpha_i * x_i\right) &\geq \sum_i \alpha_i * \varphi(x_i), \quad \text{якщо } \varphi \text{ опукла} \\
\left(\varphi\left(\sum_i \alpha_i * x_i\right) &\geq \sum_i \alpha_i * \varphi(x_i), \quad \text{якщо } \varphi \text{ ввігнута}\right)
\end{aligned}$$

$\varphi(x) = \log(x)$ – ввігнута функція, адже $\frac{d^2}{dx^2} \log(x) = -\frac{1}{x^2} < 0$, для $x \neq 0$

Тому згідно з нерівністю Єнсена має місце наступна нерівність

$$\begin{aligned}
\sum_i \alpha_i &= 1, \alpha_i \geq 0, x_i \geq 0 \\
\log\left(\sum_i \alpha_i * x_i\right) &\geq \sum_i \alpha_i * \log(x_i)
\end{aligned}$$

причому рівність досягається тоді і тільки тоді, коли $\exists i: \alpha_i = 1$

В її справедливості, а також в увігнутості функції $\varphi(x) = \log(x)$, можна переконатись безпосередньо, на рисунку 3.2.

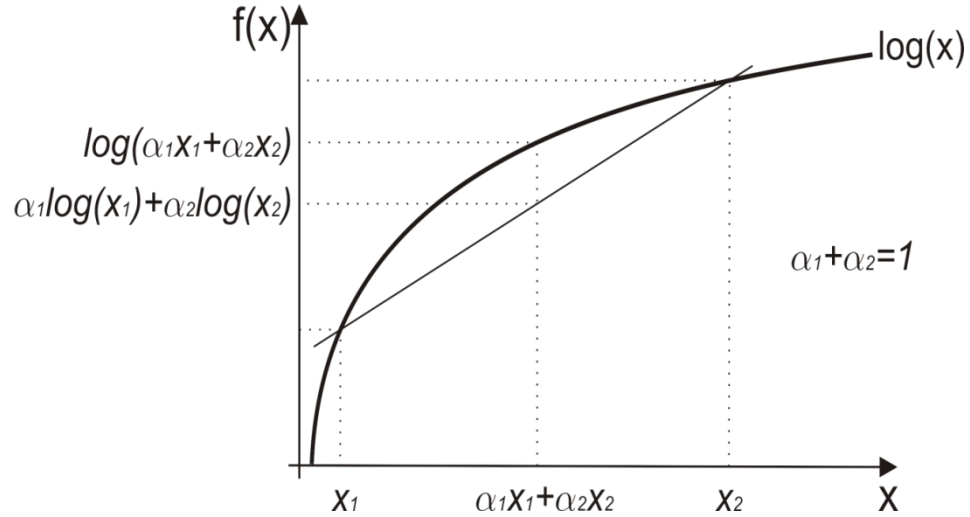


Рисунок 3.2 – Нерівність Єнсена для функції $\varphi(x) = \log(x)$

Як бачимо, в силу Нерівності Єнсена $L(\theta)$ має односторонню границю, а саме $L(\theta) \geq \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X, Y|\theta\}}{q(x)}\right)$. Надалі позначимо цю межу $Q(q, \theta)$.

Перепишемо $Q(q, \theta)$ в такому вигляді:

$$\begin{aligned} Q(q, \theta) &= \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X, Y|\theta\}}{q(x)}\right) = \\ &= \sum_x q(x) * \log(\mathcal{P}\{X, Y|\theta\}) - \sum_x q(x) * \log(q(x)) \end{aligned}$$

Як нескладно помітити другий доданок є ентропією (англ. Shannon entropy) функції $q(x)$, тому позначимо $-\sum_x q(x) * \log(q(x)) = H(q)$. При цьому перший доданок є математичним сподіванням (усередненням) $\log(\mathcal{P}\{X, Y|\theta\})$ за функцією $q(x)$. Тому також використаємо нове позначення $\sum_x q(x) * \log(\mathcal{P}\{X, Y|\theta\}) = E_{q(x)}(\log(\mathcal{P}\{X, Y|\theta\}))$. Остаточоно отримуємо в нових позначеннях:

$$\begin{aligned}
Q(q, \theta) &= \sum_x q(x) * \log(\mathcal{P}\{X, Y|\theta\}) + H(q) = \\
&= E_{q(x)} (\log(\mathcal{P}\{X, Y|\theta\})) + H(q)
\end{aligned} \tag{3.3}$$

Тепер можна побудувати ітеративний алгоритм, як модифікацію методу покоординатного спуску:

Е-крок – максимізація функції $Q(q, \theta)$, за першим параметром (розподілом $q(x)$), при фіксованому θ :

$$q^k(x) = \underset{q}{\operatorname{argmax}}(Q(q, \theta^{k-1}))$$

М-крок – максимізація функції $Q(q, \theta)$, за другим параметром (θ) при фіксованому розподілі $q(x)$:

$$\theta^k = \underset{\theta}{\operatorname{argmax}} \left(Q(q^k(x), \theta) \right) = \underset{\theta}{\operatorname{argmax}} \left(E_{q(x)} (\log(\mathcal{P}\{X, Y|\theta\})) \right)$$

Тут було опущено другий доданок, адже $H(q)$ не залежить від θ , і відповідно для максимізації можна використати лише $E_{q(x)} (\log(\mathcal{P}\{X, Y|\theta\}))$.

При цьому важливо відзначити, що Е-крок та М-крок ніколи не зменшують правдоподібність (в даному випадку логарифм функції правдоподібності):

$$L(\theta^{k-1}) = Q(q^k, \theta^{k-1}) \leq Q(q^k, \theta^k) \leq L(\theta^k)$$

Першу рівність отримуємо за Е-кроком ЕМ-алгоритму, другу нерівність маємо внаслідок максимізації на М-кроці, а третя нерівність безпосередньо випливає з нерівності Єнсена.

Продовживши аналіз цільової функції $Q(q, \theta)$ можна помітити наступне співвідношення:

$$\begin{aligned}
 Q(q, \theta) &= E_{q(x)} (\log(\mathcal{P}\{X, Y|\theta\})) + H(q) = \\
 &= \sum_x q(x) * \log(\mathcal{P}\{X, Y|\theta\}) - \sum_x q(x) * \log(q(x)) = \\
 &= \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X, Y|\theta\}}{q(x)}\right) = \\
 &= \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X|Y, \theta\} * \mathcal{P}\{Y|\theta\}}{q(x)}\right) = \\
 &= \sum_x q(x) * \log(\mathcal{P}\{Y|\theta\}) + \sum_x q(x) * \log\left(\frac{\mathcal{P}\{X|Y, \theta\}}{q(x)}\right)
 \end{aligned}$$

Тут у третьому рядку було використано формулу Баєса: $\mathcal{P}\{X, Y|\theta\} = \mathcal{P}\{Y|X, \theta\} * \mathcal{P}\{X|\theta\}$. Повертаючись до початку параграфу слід згадати що перший доданок є логарифмом функції правдоподібності: $\sum_x q(x) * \log(\mathcal{P}\{Y|\theta\}) = L(\theta)$.

Другий доданок є дивергенцією Кульбака-Лейблера (англ. Kullback-Leibler divergence). Дивергенція Кульбака-Лейблера – це невід’ємний функціонал, який є несиметричною мірою віддаленості один від одного двох імовірнісних розподілів, визначених на спільному просторі елементарних подій. Часто застосовується в теорії інформації і математичній статистиці.

У загальному випадку, якщо μ – будь-яка міра на X , для якої існують абсолютно неперервні щодо μ функції $p = \frac{dP}{d\mu}$ та $q = \frac{dQ}{d\mu}$, тоді дивергенція Кульбака-Лейблера розподілу Q щодо P визначається як:

$$D_{KL}(P||Q) = \int_X p * \log \frac{p}{q} d\mu$$

Або в дискретному випадку:

$$D_{KL}(p||q) = \sum_x p(x) * \log \left(\frac{p(x)}{q(x)} \right)$$

Хоча даний функціонал не є метрикою в просторі розподілів, оскільки не задовольняє нерівності трикутника і аксіомі симетричності, він часто розглядається як засіб вимірювання відстаней в просторі розподілів.

В даній роботі буде використано лише властивість $D_{KL}(P||Q) \geq 0$, яка безпосередньо випливає з нерівності Гібса:

Нерівність Гібса (англ. Gibbs' inequality)

Нехай $P = \{p_1, \dots, p_n\}$ – ймовірнісний розподіл. Тоді для довільного ймовірнісного розподілу $Q = \{q_1, \dots, q_n\}$ виконується наступна нерівність:

$$-\sum_{i=1}^n p_i * \log p_i \leq -\sum_{i=1}^n p_i * \log q_i$$

Причому рівність досягається тоді і тільки тоді, коли $\forall i: p_i = q_i$

Тепер маємо з нерівності Гібса легко бачити, що $D_{KL}(P||Q) \geq 0$, причому $D_{KL}(P||Q) = 0$ тоді і тільки тоді, коли $p \equiv q$.

Цей результат дає суттєві переваги при максимізації $Q(q, \theta)$:

$$Q(q, \theta) = L(\theta) - D_{KL}(q(x)||\mathcal{P}\{X|Y, \theta\}) \quad (3.4)$$

Тобто максимізація $Q(q, \theta)$ в даному разі еквівалентна мінімізації $D_{KL}(q(x)||\mathcal{P}\{X|Y, \theta\})$, а з нерівності Гібса випливає, що $\min_{q(x)} D_{KL}(q(x)||\mathcal{P}\{X|Y, \theta\}) = 0$, причому $q_{min}(x) = \mathcal{P}\{X|Y, \theta\}$.

Це дає підстави для суттєвого спрощення Е – кроку ЕМ-алгоритму. Тепер на кожному кроці замість мінімізації функції $q(x)$ стає відразу відоме аналітичне представлення $q_{min}(x) = \mathcal{P}\{X|Y, \theta\}$. Тобто k -та ітерація Е – кроку може бути записана у вигляді $q^k(x) = \mathcal{P}\{X|Y, \theta^{k-1}\}$.

Для роботи з прихованими ланцюгами маркова використовують спеціальний варіант ЕМ-алгоритму: алгоритм Баума-Велша.

Запишемо формулу (3.3), з урахуванням того, що $\mathcal{P}\{\vec{x}, \vec{z}|\mu\} = \pi_{z_0} * \prod_{t=1}^T A_{z_{t-1}}^{z_t} * B_{x_t}^{z_t}$:

$$\begin{aligned} E_{\mathcal{P}\{X|Y, \theta^*\}}(\log \mathcal{P}\{X, Y|\theta\}) &= \sum_{s=1}^{|S|} \mathcal{P}\{\vec{x}, \vec{s}|\mu\} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} = \\ &= \sum_{s=1}^{|S|} \pi_{s_0} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} + \sum_{s=1}^{|S|} \sum_{t=1}^T A_{s_{t-1}}^{s_t} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} \\ &+ \sum_{s=1}^{|S|} \sum_{t=1}^T B_{x_t}^{s_t} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} \end{aligned}$$

Так як невідомі параметри моделі $\mu = \{A, B, \Pi\}$ опинились розкиданими по трьох незалежним доданках, в методі Лагранжа можна виконувати оптимізацію по них окремо.

Як і на попередніх кроках необхідно буде визначити декілька додаткових змінних:

$$\begin{aligned} \xi(t, i, j) &= \mathcal{P}\{s_t = i, s_{t+1} = j|\vec{x}, \mu\} = \frac{\mathcal{P}\{s_t = i, s_{t+1} = j, \vec{x}|\mu\}}{\mathcal{P}\{\vec{x}|\mu\}} = \\ &= \frac{\alpha_i(t) * A_j^i * B_{x_t}^i * \beta_j(t+1)}{\sum_{m=1}^N \alpha_m(t) \beta_m(t)} = \\ &= \frac{\alpha_i(t) * A_j^i * B_{x_t}^i * \beta_j(t+1)}{\sum_{m=1}^N \sum_{n=1}^N \alpha_m(t) * A_j^i * B_{x_t}^i * \beta_n(t)} \end{aligned}$$

Ця змінна виражає імовірність що процес опиниться в стані i в момент часу t та в стані j в момент часу $t + 1$, за заданої моделі $\mu = (A, B, \pi)$ та серії спостережень \vec{x} .

$$\begin{aligned}\gamma_i(t) &= \mathcal{P}\{s_t = i, |\vec{x}, \mu\} = \\ &= \sum_{j=1}^N \mathcal{P}\{s_t = i, s_{t+1} = j | \vec{x}, \mu\} = \sum_{j=1}^N \xi(t, i, j)\end{aligned}$$

Ця змінна показує імовірність того, що процес опиниться в стані i в момент часу t , за заданої моделі μ та серії спостережень \vec{x} . Як нескладно переконалися $\gamma_i(t)$ можна отримати і як суму $\sum_{j=1}^N \xi(t, i, j)$.

Тепер, за методом множників Лагранжа, формули пошуку оптимальних параметрів моделі в нових позначеннях виглядатимуть так:

$$\tilde{\pi}_i = \mathcal{P}\{s_0 = i, |\vec{x}, \mu\} = \gamma_i(0)$$

Тут $\tilde{\pi}_i$ виражає ймовірність перебування процесу в стані i в момент часу 0.

$$\tilde{A}_j^i = \frac{\sum_{t=1}^T \mathcal{P}\{s_t = i, s_{t+1} = j | \vec{x}, \mu\}}{\sum_{t=1}^T \mathcal{P}\{s_t = i, |\vec{x}, \mu\}} = \frac{\sum_{t=1}^T \xi(t, i, j)}{\sum_{t=1}^T \gamma_i(t)}$$

Або в початкових позначеннях:

$$\tilde{A}_j^i = \frac{\sum_{t=1}^T \alpha_i(t) * A_j^i * B_{x_{t+1}}^j * \beta_j(t + 1)}{\sum_{t=1}^T \alpha_i(t) * \beta_i(t)} \quad (3.5)$$

Практично це означає що для того аби отримати \tilde{A}_j^i нам слід розділити кількість переходів (математичне сподівання) з стану i в стан j на кількість переходів з стану i .

$$\widetilde{B}_k^i = \frac{\sum_{t=1}^T \mathbb{1}\{(x_t \in k)\} * \mathcal{P}\{s_t = i, |\vec{x}, \mu\}}{\sum_{t=1}^T \mathcal{P}\{s_t = i, |\vec{x}, \mu\}} = \frac{\sum_{t=1}^T \mathbb{1}\{(x_t \in k)\} \gamma_i(t)}{\sum_{t=1}^T \gamma_i(t)}$$

Або в початкових позначеннях:

$$\widetilde{B}_k^i = \frac{\sum_{t=1}^T \mathbb{1}\{(x_t \in k)\} * \alpha_i(t) * \beta_i(t)}{\sum_{t=1}^T \alpha_i(t) * \beta_i(t)} \quad (3.6)$$

Що значить відношення математичного сподівання кількості перебувань в стані i , коли ми спостерігали за спостереженням k , до загальної кількості разів перебування процесу в стані i .

3.5 Суміш Гаусівських компонент

Ми побудували повноцінну модель прихованого марківського ланцюга, для якого тепер можливо обчислювати параметри, тобто навчати за навчальною вибіркою, робити прогнози з допомогою алгоритму Вітербі, та перевіряти степінь правдивості даних (ймовірність того що вказана послідовність спостережень відповідає моделі).

Проте ми щоразу вважаємо функції розподілу отриманих спостережень дискретними. Це цілком нормально в більшості ситуацій, навіть при дослідженні фінансових даних. Більше того, це значно спрощує моделювання досліджуваної проблеми, у тому числі й кризових явищ в економіці. Насправді значно частіше доводиться працювати з неперервними у часі

даними. В такому разі необхідно розбити функцію розподілу на частини та використати квантування початкових даних. Чудовий огляд найпопулярніших технік здійснення квантування даних представлено в роботах [3]

Наприклад при аналізі валютного ринку ми можемо ввести 5 дискретних станів: велике падіння, мале падіння, ринок без змін, мале зростання та велике зростання. Такий підхід був запропонований для прикладу в роботі [14]. Такі ж стани можна виділити і для пошуку кризових ситуацій. За період квантування прийматимемо один день, тим паче що більшість макроекономічних даних у вільному доступі мають саме такий період квантування.

Однак це не може нас задовольнити повною мірою, адже доволі часто (в роботі з реальними даними – завжди) представлених спостережень виявляється недостатньо для ефективного тренування моделі. Для розв’язання цієї проблеми нема іншого виходу, ніж використати неперервні розподіли.

При побудові моделі було вказано, що функція залежності спостережень від прихованих станів ланцюга Маркова має наступний вигляд:

$$B_{x_t}^j = \mathcal{P}\{x_t = k, s_t = j\}, \quad k \in V$$

Спробуємо представити ці функції у вигляді композицій Гаусівських розподілів з різними параметрами:

$$b_j(x_t) = \sum_{k=1}^M w_{jk} * b_{jk}(x_t), \quad \text{для } j = 1, \dots, N$$

При цьому на вагові коефіцієнти накладено наступні вимоги:

$$\begin{cases} \sum_{k=1}^M w_{jk} = 1, & \text{для } j = 1, \dots, N \\ w_{jk} \geq 0, & \text{для } j = 1, \dots, N; k = 1, \dots, M \end{cases} \quad (3.7)$$

Кожну компоненту b_{jk} будемо вважати щільністю нормальної випадкової величини з параметрами μ_{jk} та σ_{jk} .

$$b_{jk}(x_t) = N(x_t, \mu_{jk}, \sigma_{jk})$$

Тобто при обчисленні використовуватиметься формула:

$$b_{jk}(x_t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_t - \mu_{jk})^2}{2\sigma_{jk}^2}}$$

Таким чином, враховуючи що функція $\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x - \mu_{jk})^2}{2\sigma_{jk}^2}}$ є пронормованою, як щільність нормальної випадкової величини, а w_{jk} задовольняють вимогам, ми можемо гарантувати, що $b_j(x_t) = \sum_{k=1}^M w_{jk} * b_{jk}(x_t)$ теж буде коректно-визначеною щільністю деякої випадкової величини.

Теоретично такий підхід дозволяє представити будь-який розподіл у вигляді (3.11). Наприклад при суміші двох гаусівських компонент можуть бути легко отримані наступні результати, зображені на рисунку 3.3:

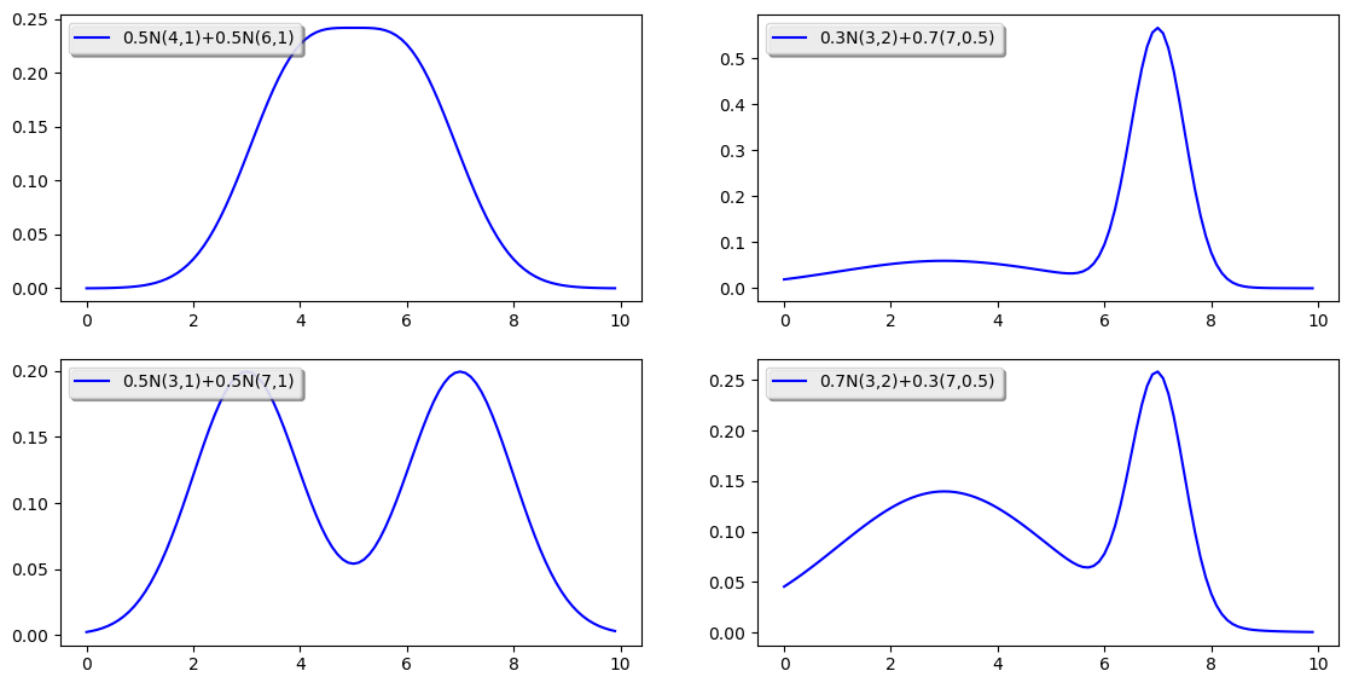


Рисунок 3.3 – Варіанти сумішей при використанні двох Гаусівських КОМПОНЕНТ

Тепер на етапі тренування моделі можливо генерувати стільки даних, скільки потрібно. Це дозволить максимально збільшити точність побудованого ланцюга. Звісно, при цьому можуть виникати помилки, пов'язані з неточностями наближення дискретного розподілу неперервним, проте вхідні дані в будь-якому разі містять неточності, відповідно ми можемо піти на такі втрати.

Слід додатково відзначити, що за таких умов ми не лише позбулися «дискретності» даних, а й вирішили іншу важливу проблему – розширення моделі. Тепер додавання нового фактора не призводитиме до створення нової моделі, а в Ланцюгу Маркова ми просто редагуватимемо кількість доданків у сумі $\sum_{k=1}^M w_{jk} * b_{jk}(x_t)$. Та перетворимо μ_{jk} та σ_{jk} на вектор та матрицю відповідно.

Зазначимо що розмірність часових рядів не повинна бути надто великою, адже це значно збільшить складність обчислень та уповільнить роботу програми, водночас ми повинні включити найбільш корельовані фактори до нашої моделі.

Зміни розрахунку компонент $b_j(x_t)$ суттєво не впливають на обчислення параметрів $\tilde{\pi}_l$ та \tilde{A}_j^l , за ЕМ алгоритмом, але тепер виникає нова група параметрів, що мають бути визначені для того аби була можливість використовувати $b_j(x_t)$ в формулах розрахунку. Якщо переписати формулу ($E_{\mathcal{P}\{X|Y,\theta^*\}}(\log \mathcal{P}\{X,Y|\theta\})$) для неперервного випадку, зміниться лише третій доданок, а саме обчислення $B_{x_t}^{z_t}$:

$$\begin{aligned} E_{\mathcal{P}\{X|Y,\theta^*\}}(\log \mathcal{P}\{X,Y|\theta\}) &= \sum_{s=1}^{|S|} \mathcal{P}\{\vec{x}, \vec{s}|\mu\} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} = \\ &= \sum_{s=1}^{|S|} \pi_{z_0} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} + \sum_{s=1}^{|S|} \sum_{t=1}^T A_{s_{t-1}}^{s_t} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} + \\ &+ \sum_{s=1}^{|S|} \sum_{t=1}^T \frac{w_{jk}}{\sqrt{2\pi}\sigma} e^{-\frac{(x_t - \mu_{jk})^2}{2\sigma_{jk}^2}} * \mathcal{P}\{\vec{s}|\vec{x}, \mu\} \end{aligned}$$

Для того, аби переписати формули для розрахунку параметрів моделі, необхідно ввести ще одну додаткову змінну:

$$\gamma_{ik}(t) = \mathcal{P}\{\tau_{it} = k, x_t = i\}$$

Ця змінна показує ймовірність того, що i -а компонента згенерувала спостереження x_t . Відповідно тут τ_{it} це випадкова величина, що визначає компонент гаусівської композиції в момент часу t для стану i . Відповідно в старих позначеннях можна записати:

$$\gamma_{ik}(t) = \mathcal{P}\{\tau_{it} = k, x_t = i\} = \gamma_i(t) * \frac{w_{ik} * b_{ik}(x_t)}{b_i(x_t)}$$

Тепер після диференціювання, за методом множників Лагранжа отримаємо наступні формули для розрахунку параметрів $b_j(x_t)$:

$$w_{ik} = \frac{\sum_{t=1}^T \mathcal{P}\{\tau_{it} = k, x_t = i\}}{\sum_{t=1}^T \mathcal{P}\{s_t = i, |\vec{x}, \mu\}} = \frac{\sum_{t=1}^T \gamma_{ik}(t)}{\sum_{t=1}^T \gamma_i(t)} \quad (3.8)$$

$$\mu_{ik} = \frac{\sum_{t=1}^T \mathcal{P}\{\tau_{it} = k, x_t = i\} x_t}{\sum_{t=1}^T \mathcal{P}\{\tau_{it} = k, x_t = i\}} = \frac{\sum_{t=1}^T \gamma_{ik}(t) x_t}{\sum_{t=1}^T \gamma_{ik}(t)} \quad (3.9)$$

$$\sigma_{ik} = \frac{\sum_{t=1}^T \mathcal{P}\{\tau_{it} = k, x_t = i\} * (x_t - \mu_{ik})^2}{\sum_{t=1}^T \mathcal{P}\{\tau_{it} = k, x_t = i\}} = \frac{\sum_{t=1}^T \gamma_{ik}(t) (x_t - \mu_{ik})^2}{\sum_{t=1}^T \gamma_{ik}(t)} \quad (3.10)$$

Таким чином на кожній ітерації ЕМ-алгоритму ми отримуватимемо покращені функції розподілу $b_j(x_t)$, і вже потому генерувати тренувальну послідовність спостережень:

$$b_j(x_t) = \sum_{k=1}^M w_{jk} * b_{jk}(x_t)$$

$$b_{jk}(x_t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_t - \mu_{jk})^2}{2\sigma_{jk}^2}} \quad (3.11)$$

Далі, застосовуючи Алгоритм Баума-Велша, отримаємо оптимальні параметри моделі:

$$\tilde{\pi}_i = \gamma_i(0)$$

$$\tilde{A}_j^i = \frac{\sum_{t=1}^T \xi(t, i, j)}{\sum_{t=1}^T \gamma_i(t)} = \frac{\sum_{t=1}^T \alpha_i(t) * A_j^i * B_{x_{t+1}}^j * \beta_j(t+1)}{\sum_{t=1}^T \alpha_i(t) * \beta_i(t)} \quad (3.12)$$

Традиційний ЕМ-алгоритм добре проявив себе у роботі з розпізнаванням мови, сигналів, аналізом аудіо, відео та текстового контенту. Спершу цей алгоритм був строго формалізований в 1977 році [13] загальному вигляді, та може бути використаний не лише у прихованих ланцюгах Маркова, а просто в моделях де присутні приховані змінні та змінні, що спостерігаються. Але в його формулюванні є одна суттєва деталь.

Справа в тому, що традиційні варіанти використання ЕМ алгоритму не враховують різну важливість даних у вибірці. В базовому випадку висувається припущення, що всі елементи тренувальної вибірки мають однаковий вплив на прогнозовані значення, та відповідно всі спостереження в спостережуваній послідовності матимуть однакова вагу. Цей підхід добре працює, та є цілком логічним у роботі з задачами розпізнавання мови. В такому випадку, з точки зору аналізу всі спостереження є однаковими, та повинні мати однаковий вплив на результат навчання моделі.

Однак при роботі з фінансовими даними ситуація стає цікавішою. Справа в тому, що на відміну від розпізнавання сигналів, послідовність фінансових даних має неоднакову «важливість» у часі. Так, ми повинні пам'ятати, що нещодавно отримані дані мають значно більшу вагу, аніж ті, що були отримані кілька років тому. Хорошим прикладом цьому є індекс Доу Джонса (Dow Jones Index). Індекс Доу Джонса зріс з 200 до 2000 за 57 років, проте після цього виріс з 2000 до 10000 всього за 12 років. Це знову ж таки засвідчує, що у фінансових даних більшу вагу мають ті дані, що спостерігалися пізніше.

Для врахування різної важливості спостережень, додамо часову функцію η_t для обчислення правдоподібності серії спостережень:

$$\tilde{\mathcal{P}}\{x_t|\mu\} = \eta_t * \mathcal{P}\{x_t|\mu\}$$

Тепер в якості функції $\eta_t = \eta(t)$ підібравши таку, яка б переважала для останніх спостережень, завершимо побудову моделі.

При цьому в такого підходу є суттєвий недолік. При використанні зважених значень, а особливо експоненційно-зважених, експоненційно збільшується чутливість моделі до різких змін. Іноді для того, аби нівелювати (або принаймні зменшити) вплив стрибків на експоненційні показники часто застосовують подвійний зважений алгоритм (double weighted EM algorithm) [12], в інших же випадках просто відмовляються від концепції використання зважених величин.

На щастя, в даній роботі предметом дослідження є виявлення кризових явищ в економіці, тобто виявлення саме тих стрибків, після яких останній підхід даватиме некоректні результати. Тому в такому разі можна спробувати скористатись слабкою стороною зваженого алгоритму, та перетворити її на сильну. Змоделюємо поведінку економіки в моделі:

$$\left\{ \begin{array}{l} s_1 = \text{велике падіння;} \\ s_2 = \text{мале падіння} \\ s_3 = \text{ринок без змін} \\ s_4 = \text{мале зростання} \\ s_5 = \text{велике зростання} \end{array} \right\}$$

Тепер можна не хвилюватися, що різкі переходи в стани s_5 та s_1 серйозно спотворюватимуть результати. Справді, перехід в стан s_1 – це те що необхідно виявляти в моделі, а через це модель навпаки працюватиме навіть краще, адже зможе виявляти кризові явища на декілька моментів часу раніше. Інший перехід (s_5) становить загрозу, адже додає до моделі сліпу зону: вона тепер не зможе вчасно виявити стрімке падіння після стрімкого зростання, однак такі ситуації при нормальному функціонуванні економіки виникають рідко, а якщо подібна ситуація виникла в кризовий період, то не

вплине на момент виявлення кризи, хоча й може серйозно спотворити прогноз щодо виходу з критичної ситуації

Таке ускладнення моделі могло б значно ускладнити пошук оптимальних параметрів, адже змінюється всі розрахунки правдоподібності. Зокрема у виразі для максимізації з'явиться нова функція під логарифмом:

$$\begin{aligned}
 E_{\mathcal{P}\{X|Y,\theta^*\}}(\log \eta \mathcal{P}\{X, Y|\theta\}) &= \sum_{z=1}^{|S|} \log \mathcal{P}\{\vec{x}, \vec{z}|\mu\} * \mathcal{P}\{\vec{z}|\vec{x}, \mu\} = \\
 &= \sum_{z=1}^{|S|} \log \pi_{z_0} * \mathcal{P}\{\vec{z}|\vec{x}, \mu\} + \sum_{z=1}^{|S|} \sum_{t=1}^T \log A_{z_{t-1}}^{z_t} * \mathcal{P}\{\vec{z}|\vec{x}, \mu\} + \\
 &+ \sum_{z=1}^{|S|} \sum_{t=1}^T \frac{w_{jk}}{\sqrt{2\pi}\sigma} e^{-\frac{(x_t - \mu_{jk})^2}{2\sigma_{jk}^2}} * \mathcal{P}\{\vec{z}|\vec{x}, \mu\}
 \end{aligned}$$

Тепер, після перерахунку Q -функції з урахуванням часового вагового коефіцієнта η_t коефіцієнти, що необхідні для визначення $b_j(x_t) = \sum_{k=1}^M w_{jk} * b_{jk}(x_t)$ можуть бути отримані з наступних формул:

$$\mu'_{ik} = \frac{\sum_{t=1}^T \eta_t \gamma_{ik}(t) x_t}{\sum_{t=1}^T \eta_t \gamma_{ik}(t)}$$

Для математичного сподівання:

$$\sigma'_{ik} = \frac{\sum_{t=1}^T \eta_t \gamma_{ik}(t) (x_t - \mu'_{ik})^2}{\sum_{t=1}^T \eta_t \gamma_{ik}(t)}$$

Для дисперсії:

$$w'_{ik} = \frac{\sum_{t=1}^T \eta_t \gamma_{ik}(t)}{\sum_{t=1}^T \eta_t \gamma_i(t)}$$

При цьому розрахунок матриці перехідних імовірностей на М-етапі ЕМ алгоритму (в даному випадку при застосуванні алгоритму Баума-Велша) обчислюватиметься таким чином:

$$\hat{A}_j^t = \frac{\sum_{t=1}^T \eta_t \alpha_i(t) * A_j^i * B_{x_{t+1}}^j * \beta_j(t+1)}{\sum_{t=1}^T \eta_t \alpha_i(t) * \beta_i(t)}$$

3.6 Нормалізація моделі

При обчисленні параметрів моделі активно використовуються $\alpha_i(t)$, $\beta_j(t+1)$ та їхні добутки. При цьому в ході обчислень вони можуть виявитись дуже малими (особливо $\alpha_i(t)$ для великих t та $\beta_j(t+1)$ для малих $[7]$). При написанні програми це призводитиме до виходів за межі машинної точності, або ділення на нуль. При цьому легко помітити, що в формулах розрахунку для \tilde{A}_j^t та \tilde{B}_k^t (в неперервному випадку \tilde{A}_j^t , w_{ik} , μ_{ik} , σ_{ik}) дані вирази входять і в чисельник і в знаменник. Саме тому якщо для помножити ці значення на одне і те ж число, це не вплине на подальші розрахунки.

Впринципі таким числом можна зробити довільну наперед задану константу, але враховуючи що для різних t порядок $\alpha_i(t)$ може відрізнятись від 10^{-2} до 10^{-100} , варто використати гнучкіший підхід. Для кожного t обиратимемо індивідуальне, співвимірне з $\alpha_i(t)$ число $p(t)$. Для таких цілей ідеально підходить ймовірність спостереження послідовності $\{x_1, x_2, \dots, x_t\}$: $\mathcal{P}\{x_1, x_2, \dots, x_t, A, B\} = \sum_{i=1}^{|S|} \alpha_i(t)$.

Тепер ітерацію ЕМ-алгоритму представимо наступним чином:

1. Forward Procedure (ітерації від $t = 1$ до $t = T$)
2. Backward Procedure (ітерації від $t = T$ до $t = 1$)
3. Для кожного t обчислення $p(t) = \sum_{i=1}^{|S|} \alpha_i(t)$
4. Для кожного t позначимо $\widehat{\alpha}_i(t) = \frac{\alpha_i(t)}{p(t)}$, $\widehat{\beta}_i(t) = \frac{\beta_i(t)}{p(T-t)}$
5. Завершення ітерації з значеннями $\widehat{\alpha}_i(t)$, $\widehat{\beta}_i(t)$

Такої нормалізації зазвичай цілком вистачає для зручної роботи з представленими моделями, проте іноді все ж отримувані ймовірності будуть надто малі, та помилка ділення на нуль залишатиметься. В таких випадках необхідно вводити нові метрики, та використовувати техніку, описану в [12]. При обчисленні ймовірностей (2.11) теж можуть виникати обчислювальні труднощі, в ситуаціях, коли $\sigma_{ik} \rightarrow 0$. В такому разі слід обмежити рух алгоритму по координатам σ_{ik} . Для одновимірного випадку умовою $\sigma_{ik} > \epsilon$, а для багатовимірного $\det\{\sigma_{ik}\} > \epsilon$.

Висновки до розділу

У цьому розділі було побудовано математичну модель - прихований ланцюг Маркова, який дозволяє приховувати реальні стани та діяти в припущенні що спостереження ведуться за функціями над ними. Ймовірності В в цьому випадку записуються в вигляді матриці, що спростило пошук параметрів моделі за ЕМ алгоритмом. Було показано неспроможність класичного методу для прогнозування – алгоритму Вітербі для такого класу задач, та виведено власну схему здійснення прогнозів.

Було створено розширення попередньої моделі на неперервні розподіли з Гаусівськими компонентами, та перехід від матричного запису до функціонального.

Було знайдено нові формули для тренування моделі на основі ЕМ-алгоритму, та продемонстровано гнучкість моделі при додаванні нових індикаторів.

Після побудови моделей проведено огляд обчислювальних проблем, що можуть виникати при роботі з моделями, а також наведено способи їх вирішення та нормалізації моделі.

РОЗДІЛ 4 РЕЗУЛЬТАТИ РОБОТИ ПРОГРАМНОГО ПРОДУКТУ

4.1 Моніторинг соціальних мереж для визначення користувачів з ризиком депресії.

В цьому розділі буде представлено автоматизовану систему для виявлення людей в стані депресії на основі їхньої публічної соціальної активності.

Соціальною мережею для перевірки було обрано Twitter. На це є декілька причин:

1. Twitter має зручний API для доступу;
2. У Twitter повідомлення користувачів жорстко стандартизовані, та мають довжину не більше 140 символів;
3. У вільному доступі існують підготовані набори твітів за різними темами;
4. Практично всі відомі люди активно ведуть власний Twitter аккаунт.

Для досягнення поставленої цілі необхідно побудувати 2 класифікатори: Tweet-Level Classifier та User-Level Classifier.

Мета першого класифікатора передбачати чи є даний твіт проявом депресії, чи ні. Для розпізнавання було використано 2 англійські словники [19], які містили поляризацію слів та словосполучень, тобто були розмічені як позитивні, негативні, дуже позитивні та дуже негативні (незалежні терми були виключені зі словників).

Для кожного повідомлення алгоритм шукав паттерни співпадіння в словниках, та ставив у відповідність словам їх поляризацію. Загальний результат для повідомлення можна далі отримати, просумувавши всі слова у твіті.

Приклад подібного розпізнавання, для депресивного речення, зображено на рисунку 4.1:

@nabilaqila: i am so screwed up! a confused and dumb child! i NEVER felt so bad all my life! i wish i have any specialty! too bad i have none =(

Рисунок 4.1 – Розпізнавання рівня депресії в повідомленні

Хоча поляризація слів була досить очевидною, але до найпопулярніших слів в депресивних повідомлень часто потрапляють дуже несподівані кандидати: *my*, *me*, *love*, *people*, з яких можна скласти безліч позитивних речень. Узагальнену статистику за словами в депресивних твітах наведено на рисунку 4.2:

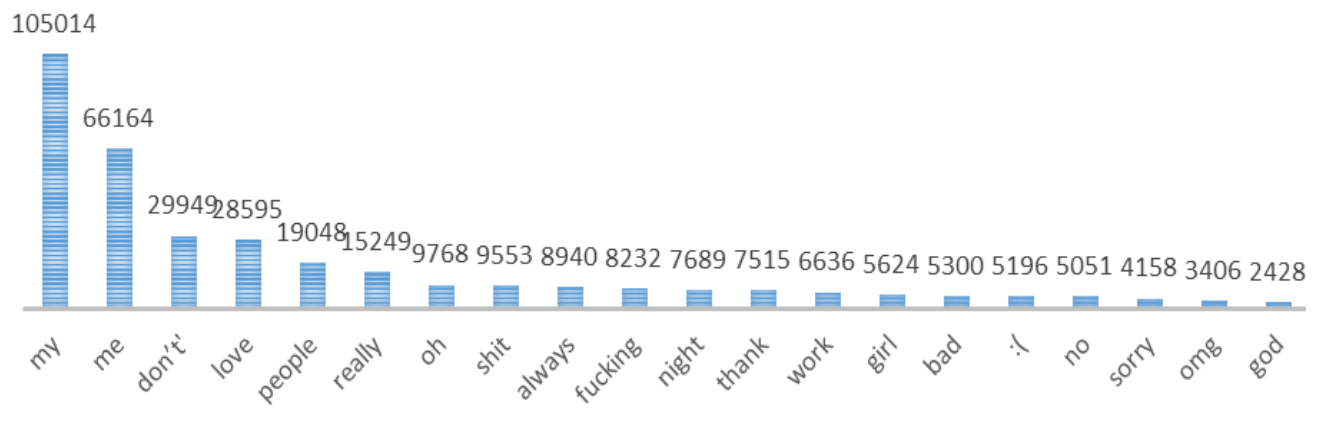


Рисунок 4.2 – Частота слів в депресивних повідомленнях.

Для перевірки класифікатора використаємо три класичні метрики: precision, recall та F1(F-Measure).

Precision – точність класифікатора, характеризує те, як багато повідомлень, ідентифікованих як депресивні насправді є депресивними.

Recall характеризує який відсоток депресивних користувачів ми насправді ідентифікували.

Метрика F1 є середнім гармонійним precision та recall:

$$F_1 = \frac{2 * precision * recall}{precision + recall}$$

Для розміченої бази в 1 600 000 повідомлень (800 000 позитивних та 800 000 негативних) було отримано результати, представлені в таблиці 4.1:

Таблиця 4.1 Обчислені метрики для класифікатора повідомлень

Метрика	Результат
<i>precision</i>	0.8905
<i>recall</i>	0.73939
F_1	0.8079

Метою побудови User-Level класифікатора буде передбачення того, чи заданий користувач страждає на депресію. При цьому джерелом даних буде його нещодавня діяльність в соціальній мережі. Для реалізації цієї мети використовуватимемо прихований Марківський ланцюг з трьома станами : {користувач схильний до депресії, користувач у переддеприсивному стані, стан користувача позитивний}. Спостережуваними змінними буде рівень депресії твітів, а вхідною послідовністю – власне повідомлення користувача.

Схема побудованої система представлена на рисунку 4.3:

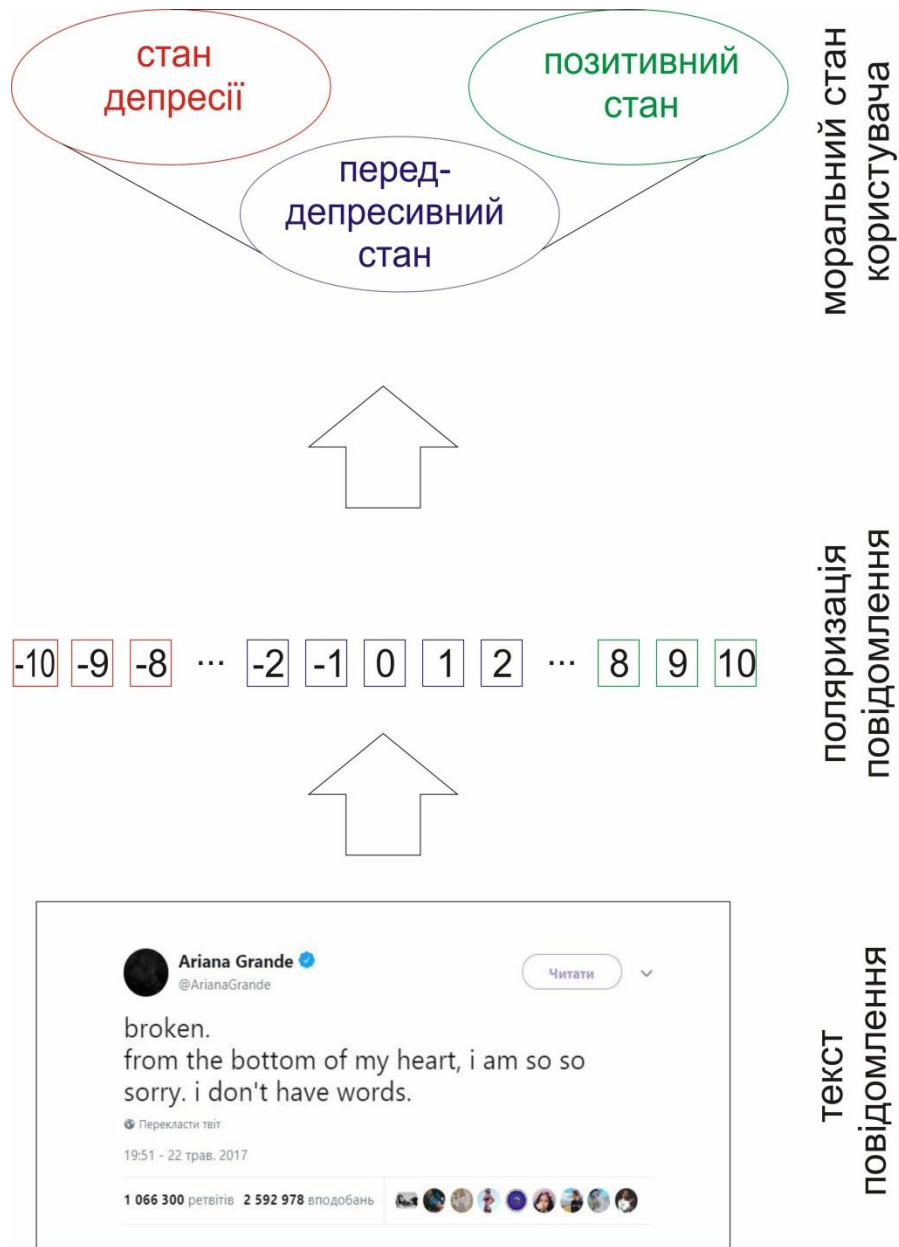


Рисунок 4.3 – Система моніторингу рівня депресії користувачів соціальної мережі Twitter.

Нарешті, розглянемо результати роботи системи раннього виявлення для окремих відомих публічних персон. Розпочнемо з добре відомих акаунтів двох останніх президентів США – Дональда Трампа та Барака Обами.

Результати роботи класифікатора для экс американського президента представлені на рисунку 4.4:

Рівень депресивності користувача Barack Obama в соціальній мережі Twitter

Аккаунт @BarackObama

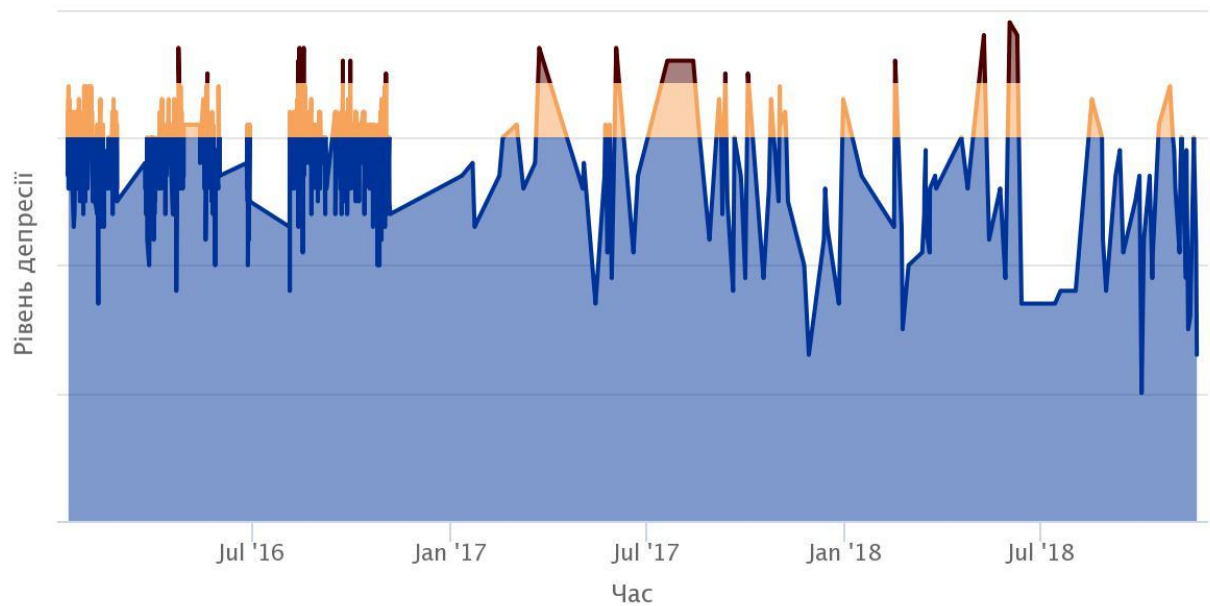


Рисунок 4.4 – Моніторинг соціальної активності Барака Обами в соціальній мережі Twitter.

Як бачимо, хоча й присутній деякий тренд на збільшення рівня депресії, риторика президента часто змінювала свої полярність. В результаті роботи програмного засобу було отримано наступні характеристики моделі, представлені у таблицях 4.2-4.4.

Таблиця 4.2 – Матриця перехідних ймовірностей

	s_0	s_1	s_2
s_0	0.502378	0.0901444	0.4074776
s_1	0.4567485	0.0585982	0.4846532
s_2	0.502378	0.0901444	0.4074776

Таблиця 4.3 – Математичні сподівання Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
μ_1	0.1471068	-2.3053496	-4.8155079

Таблиця 4.4 – Дисперсії Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
σ_1^2	5.1001334	4.5624261	33.9508057

.Далі розглянемо більш одіозного нинішнього президента Сполучених Штатів –Дональда Трампа.

Результати роботи Tweet-Level класифікатора для Дональда Трампа наведені на рисунку 4.5:

Рівень депресивності користувача Donald J. Trump в соціальній мережі Twitter

Аккаунт @realDonaldTrump

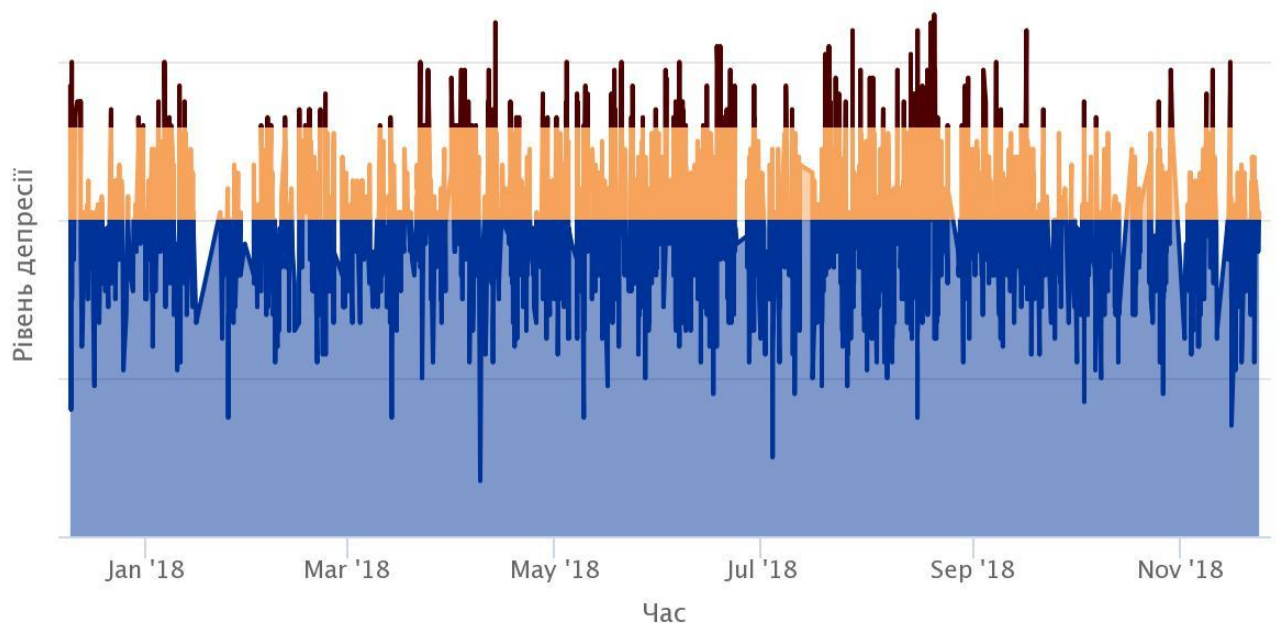


Рисунок 4.5 – Моніторинг соціальної активності Дональда Трампа в соціальній мережі Twitter.

Як бачимо, принаймні на перший погляд рівень деприсивності в повідомленнях американського лідера видається абсолютно випадковим та хаотичним. Аналіз моделювання для цього користувача дозволив отримати наступні результати, відображені в таблицях 4.5-4.7

Таблиця 4.5 – Матриця перехідних ймовірностей

	s_0	s_1	s_2
s_0	0.4957267	0.408427	0.0958462
s_1	0.2514296	0.4885628	0.2600076
s_2	0.1993208	0.4002061	0.4004732

Таблиця 4.6 – Математичні сподівання Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
μ_1	-7.4511104	-1.2360857	7.5868936

Таблиця 4.7 – Дисперсії Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
σ_1^2	42.3026352	17.6183491	43.5174294

Також можна порівняти такі результати з аккаунтом відомого футболіста Кріштіано Роналдо.

Його соціальна активність у часовому розрізі відображена на рисунку 4.6:

Рівень депресивності користувача Cristiano Ronaldo в соціальній мережі Twitter

Аккаунт @Cristiano

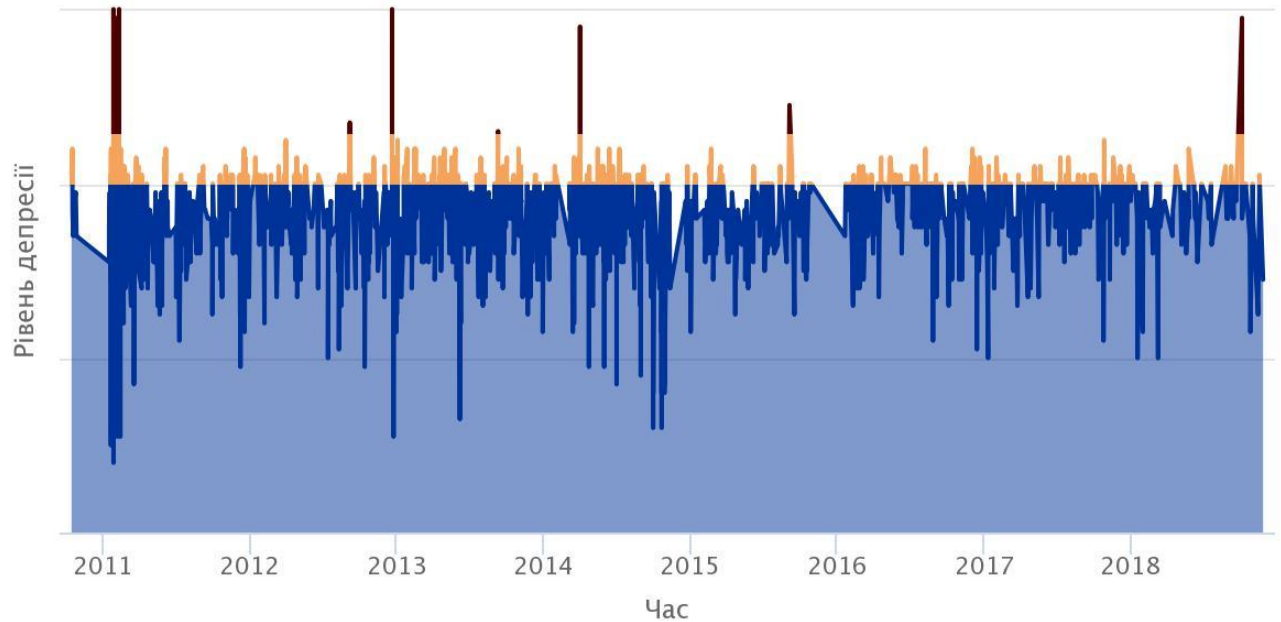


Рисунок 4.6 – Моніторинг соціальної активності Кріштіано Роналдо в соціальній мережі Twitter.

Результати оцінки соціальної оцінки п'ятикратного володаря золотого м'яча відображені в таблицях 4.8-4.10:

Таблиця 4.8 – Матриця перехідних ймовірностей

	s_0	s_1	s_2
s_0	0.557193	0.3715683	0.0712387
s_1	0.5159044	0.3761161	0.1079796
s_2	0.3653862	0.3742364	0.2603774

Таблиця 4.9 – Математичні сподівання Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
μ_1	-1.0971193	-5.7664218	-9.8810701

Таблиця 4.10 – Дисперсії Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
σ_1^2	4.5800405	11.9221182	109.2463379

Позитивний тренд в акаунті португальця настільки очевидний, що його можна порівняти лише з відомою американською комедіанткою - Еллен Лі Дедженерес. Позитивність повідомлень, які були опубліковані з акаунту Еллен Лі Дедженерес відображено на рисунку 4.7

Рівень депресивності користувача Ellen DeGeneres в соціальній мережі Twitter

Акаунт @TheEllenShow

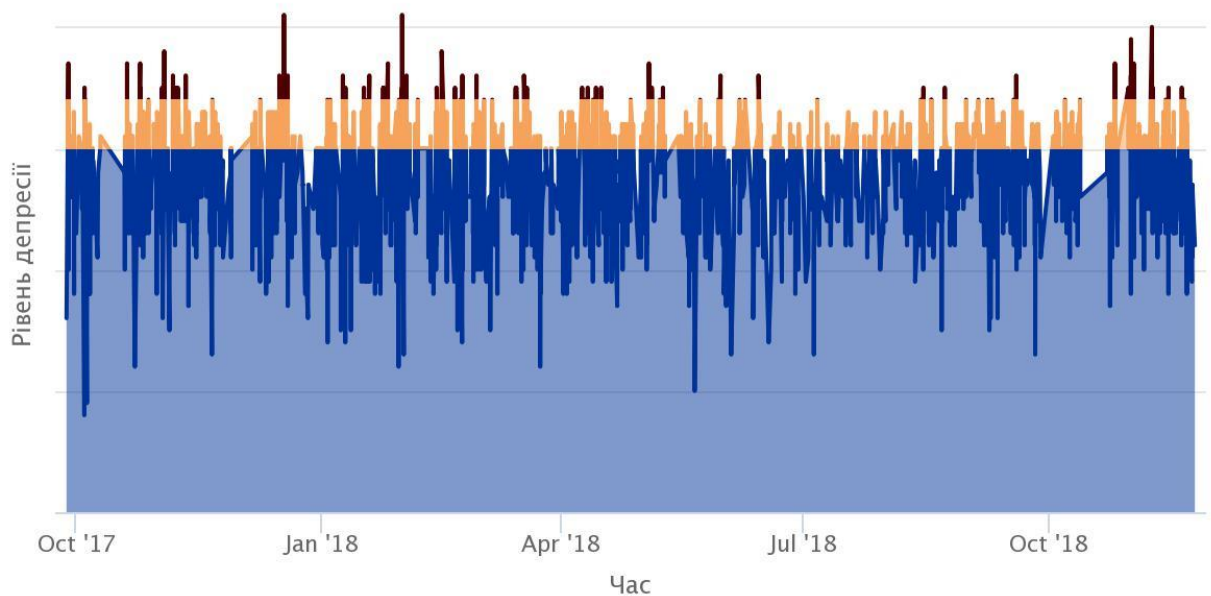


Рисунок 4.7 – Моніторинг соціальної активності Еллен Лі Дедженерес в соціальній мережі Twitter.

Результати алгоритму оцінювання наведені в таблицях 4.11-4.13:

Таблиця 4.11 – Матриця перехідних ймовірностей

	s_0	s_1	s_2
s_0	0.3254468	0.4285387	0.2460146
s_1	0.2349717	0.5357559	0.2292724
s_2	0.2470246	0.5605528	0.1924226

Таблиця 4.12 – Математичні сподівання Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
μ_1	-1.0971193	-5.7664218	-9.8810701

Таблиця 4.13 – Дисперсії Гаусівських сумішей (одновимірний випадок)

	s_0	s_1	s_2
σ_1^2	4.5800405	11.9221182	109.2463379

Цікаво, що на противагу звичайним користувачам можна представити аккаунти світових ЗМІ, які щоденно доносять в мережу світові новини. І, що цілком прогнозовано, рівень депресивності в їхніх повідомленнях значно перевищує будь-які пороги депресивності.

Нижче, на рисунках 4.8-4.10 наведено результати прогнозування для найбільших світових ЗМІ: BBC, CNN та The New York Times.

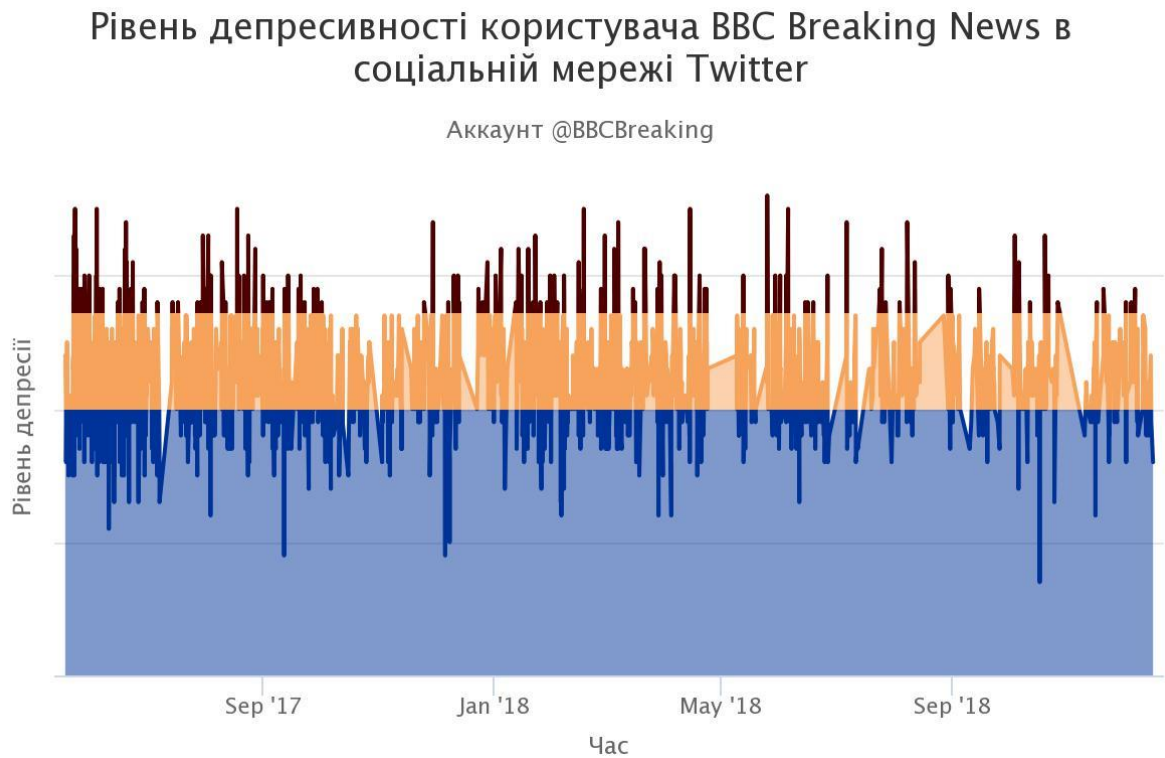


Рисунок 4.8 – Моніторинг соціальної активності BBC в соціальній мережі Twitter.

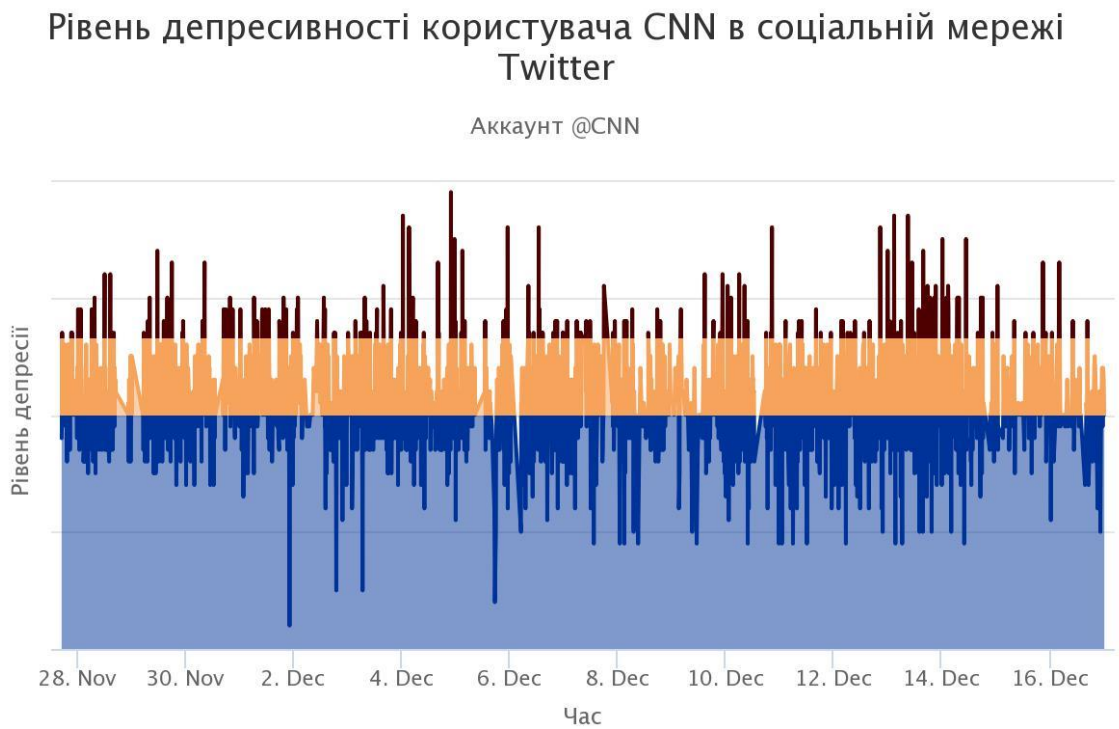


Рисунок 4.9 – Моніторинг соціальної активності CNN в соціальній мережі Twitter.

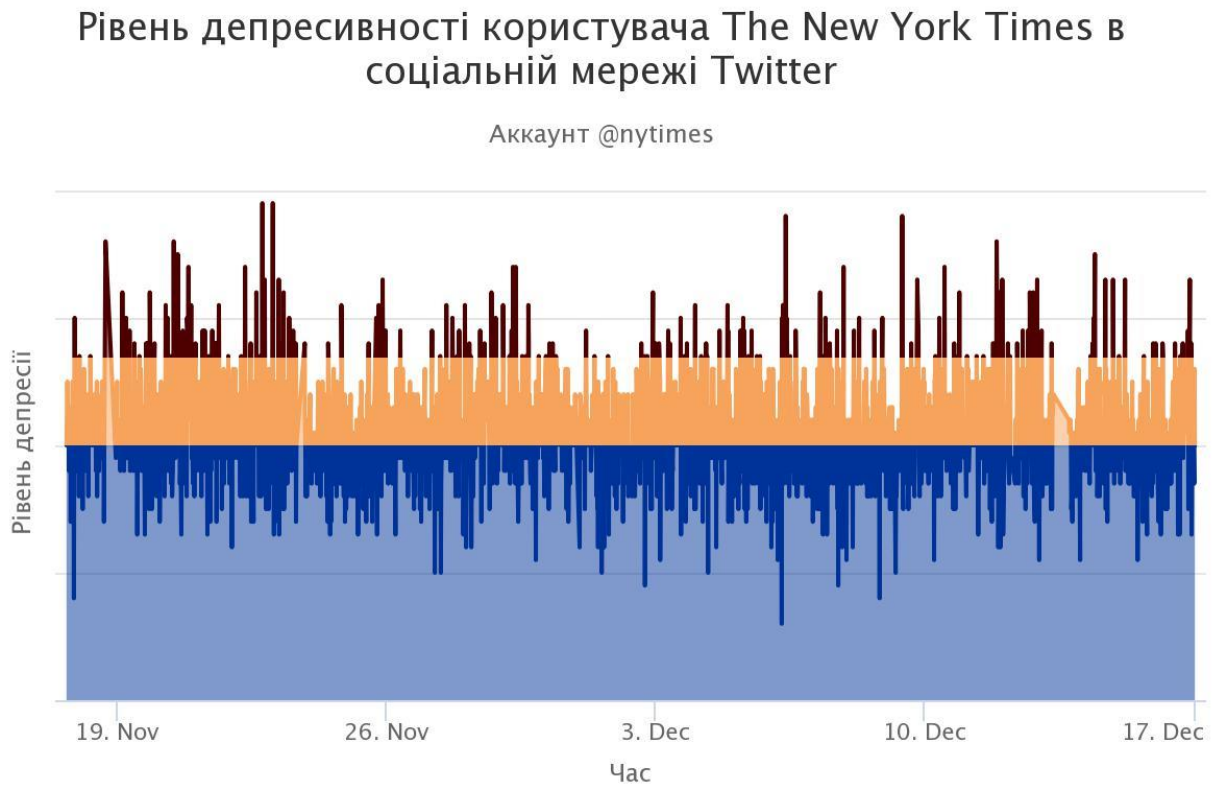


Рисунок 4.9 – Моніторинг соціальної активності CNN в соціальній мережі Twitter.

Порівняти результати для всіх досліджених аккаунтів можна у таблиці 4.14:

Таблиця 4.14 – Порівняння результатів розпізнавання та прогнозування рівня депресії найпопулярніших користувачів соціальної мережі Twitter.

Користувач	Кількість підписників	Період спостереження	Негативних повідомлень	Ймовірність депресивного твіту
Barack Obama	103,600,058	13/01/2016 21/11/2018	151/656	0.1602357587 7383198

Продовження таблиці 4.14

Користувач	Кількість підписників	Період спостереження	Негативних повідомлень	Ймовірність депресивного твіту
Donald J. Trump	56,259,939	09/12/2017 24/11/2018	886/2325	0.3217525861 5398983
Cristiano Ronaldo	75,502,644	13/10/2010 25/11/2018	216/1761	0.0700637747 5062958
Ellen DeGeneres	76,689,215	27/09/2017 24/11/2018	499/2165	0.1511605580 4128878
BBC Breaking News	38,591,826	18/05/2017 15/12/2018	1245/1816	0.5080383535 665334
CNN	40,897,892	27/11/2018 17/12/2018	1226/2357	0.4274146960 120635
The New York Times	42,453,086	17/10/2018 17/11/2018	1204/2216	0.4833157778 8901825

Висновки до розділу

В цьому розділі було побудовано автоматизовану систему для виявлення людей в стані депресії на основі їхньої публічної соціальної активності. Було показано роботу побудованої моделі на найпопулярніших акаунтах соціальної мережі Twitter, обчислено основні метрики перевірки адекватності запропонованих класифікаторів.

РОЗДІЛ 5 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

5.1 Опис ідеї проекту

Метою проекту є залучення прихованих марківських ланцюгів до проблеми своєчасного виявлення кризових явищ.

Опис ідеї стартап-проекту подано в таблиці 5.1.

Таблиця 5.1 - Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розпізнавання кризових явищ з допомогою прихованих Марківських ланцюгів	Управління людськими ресурсами	Збільшення контролю над станом працівників
	Управління запасами	Вчасна закупка актуальних товарів
	Моніторинг фінансових ризиків	Своєчасна реакція на зміну кон'юктури ринку
	Управління технологічним виробництвом	Контроль над важливими характеристиками виробництва

Для реалізації проекту необхідні наступні ресурси:

- Доступ до реальних даних, попередньо оброблених даних

- Підписка на наукові видання в галузі розпізнавання кризових явищ та прихованих Марківських ланцюгів
- Можливість залучення наукових інститутів
- Комп'ютери для розробки ПЗ
- Тестові комп'ютери та мобільні пристрої для тестування ПЗ

Сильні, слабкі та нейтральні характеристики ідеї проекту зображено в таблиці 5.2.

Таблиця 5.2 - Визначення характеристик ідеї проекту

№ п/п	Техніко- економічні характеристик и ідеї	Потенційні) товари/концепції конкурентів			W (слабка сторона)	N (нейтр. сторона)	S (сильна сторона)
		FS	Sales Loft	lan. Analyt P			
1	Універсальний формат даних	-	+	-	+		
2	АВС-аналіз	+	+	-			+
3	Асоц. аналіз	+	-	+			+
5	Моделювання і прогнозування	+	+	+		+	
6	Відомість бренду	+	-	-	+		

5.2 Технологічний аудит ідеї проекту

Технологічний аудит ідеї проекту наведений у таблиці 5.3.

Таблиця 5.3 - Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Алгоритм Баума-Велша	Python, JS	Технологія наявна Потрібно оптимізувати алгоритм.	Технологія загальнодоступна
2	Forward Procedure	R, Python, JS	Технологія наявна і не потребує змін.	Технологія загальнодоступна
3	Backward Procedure	R, Python, JS	Технологія наявна і не потребує змін.	Технологія загальнодоступна
4	Процедура прогнозування	Python	Необхідна реалізація	Технологія не адаптована під поставлену задачу
Обрана технологія реалізації ідеї проекту: для реалізації проекту обрана мова програмування JavaScript.				

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Характеристика потенційного ринку стартап-проекту наведена у таблиці 5.4.

Таблиця 5.4 - Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	4
2	Загальний обсяг продаж, грн/ум.од	1000 ум.од
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	20%

Характеристика потенційних клієнтів наведена в таблиці 5.5.

Таблиця 5.5 - Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці цільових груп клієнтів	Вимоги споживачів до товару
1	Прогнозування та передбачення потреб користувачів у роздрібній торгівлі	Середній та великий бізнес, що застосовують системи керування ресурсів підприємства	ERP система підприємства, розміри оброблюваних даних, технічні обмеження	Ефективність прогнозування Швидка обробка даних Оптимальне використання ресурсів

Можливі загрози для стартап-проекту наведені у таблиці 5.6.

Таблиця 5.6 - Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Невпорядкованість і неповнота даних	Клієнтські бази можуть містити невпорядковані дані і також певні дані можуть бути відсутніми	Додавання модуля попередньої обробки даних
2	Нестача технічних ресурсів	Клієнти можуть мати обмежені локальні технічні ресурси, недостатні для повноцінної роботи системи	Винесення модуля обчислення на сервери компаній-партнерів

Фактори можливостей наведені у таблиці 5.7.

Таблиця 5.7 - Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Хмарні обчислення	Можливість виконання усіх обчислень на віддалених серверах	Пристосування модулів обчислення для роботи на сервері

Продовження таблиці 5.7

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
2	Коригування прогнозу	Можливість коригування прогнозу в режимі реального часу на основі власної бази даних та спорідненої інформації з інтернету	Розробка модулів інтеграції з google analytics, оптимізація застосунку для можливості роботи в реальному часі

Проведений ступеневий аналіз конкуренції на ринку зображено у таблиці 5.8.

Таблиця 5.8 - Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - олігополія	Присутня невелика кількість фірм. Більшість ринку контролюють фірми-гіганти	Впровадження технологічних інновацій. Кооперація з дослідницькими центрами.

Продовження таблиці 5.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
2. За рівнем конкурентної боротьби - глобальний	Продукція не залежить від країни чи локалізації клієнта	Розширення функціоналу та задоволення потреб клієнтів.
3. За галузевою ознакою внутрішнього лузева	Продукт спрямований на роздрібну торгівлю	
5. Конкуренція за видами товарів: - за бажанням	Полягає у випередженні задоволення бажань клієнта	
5. За характером конкурентних переваг - нецінова	Переваги передбачають собою ефективність та різноманіття функціоналу	
6. За інтенсивністю - не марочна	Торгова марка майже немає впливу	

Проведений аналіз конкуренції в галузі зображено у таблиці 5.9.

Таблиця 5.9 - Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти
	SalesLoft, IBM, Planning Analytics	SlickPie
Висновки	Контролюють велику частину ринку, мають узагальнені рішення	Спрямовані на малий бізнес, не мають локалізацій для більшості країн Європи

Фактори конкурентоспроможності та їх обґрунтування наведені в таблиці 5.10.

Таблиця 5.10 - Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Інновації	Інноваційні рішення мають забезпечити перевагу нашим клієнтам над конкурентами
2	Функціонал	Функціонал повинен покривати вирішення необхідних задач клієнтів
3	Цінова політика	Вартість продукту відіграє велику роль при виборі системи клієнтом
4	Ресурсоемність	Великі затрати технічних ресурсів можуть спровокувати необхідність залучення додаткових коштів

Порівняльний аналіз сильних та слабких сторін проекту відображено у таблиці 5.11.

Таблиця 5.11 - Порівняльний аналіз сильних та слабких сторін RFS

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з RFS						
			3	2	1		1	2	3
1	Інновації	2							
2	Функціонал	8							
3	Цінова політика	7							
4	Ресурсоемність	0							

SWOT-аналіз проекту наведено в таблиці 5.12.

Таблиця 5.12 - SWOT-аналіз стартап-проекту

Сильні сторони: розумна цінова політика, функціонал забезпечує рішення більшості задач клієнта	Слабкі сторони: відсутність співпраці з інноваційними центрами, велика ресурсоемність, можливі проблеми при нечіткості та високій волатильності даних
Можливості: впровадження інноваційних рішень, оптимізація роботи продукту	Загрози: неоптимізована робота продукту, узагальнені рішення, що непридатні для вирішення клієнтських задач

Альтернативи ринкового впровадження проекту розглянуто в таблиці 5.13.

Таблиця 5.13 - Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Спеціалізовані рішення	Висока	6-9 місяців
2	Хмарний сервіс	Висока	9-12 місяців
3	Узагальнення рішення, вихід на нові сфери ринку	Середня	12-15 місяці
4	Створення SAAS рішення, перенесення обчислень на сервери клієнтів	Низька	15-36 місяців

5.4 Розроблення ринкової стратегії проекту

Опис та вибір цільових груп потенційних клієнтів зображено в таблиці 5.14.

Таблиця 5.14 - Вибір цільових груп потенційних споживачів

№ п/п	Опис цільової групи потенційн их клієнтів	Готовність споживачів сприйняти продукт	Орієнто вний попит в межах цільової групи	Інтенсивн ість конкурен ції в сегменті	Простота входу у сегмент
1	Малий бізнес	Низька готовність, через необхідність вкладання великих коштів.	Низький попит	Середня	Вхід в сегмент складний
2	Середній бізнес	Середня готовність. В залежності від виду бізнесу, готовність різниться.	Середні й попит	Вище середньої	Вхід в сегмент достатньо складний
3	Великий бізнес	Абсолютна готовність. Більшість компаній такого рівня застосовували чи планують застосовувати інтелектуальні системи прогнозування, тому повністю готові розглянути альтернативні рішення	Високий попит	Середня	Вхід в сегмент складний
Які цільові групи обрано: 2,3					

В таблиці 5.15 зображено вибір базової стратегії розвитку.

Таблиця 5.15 - Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Розробка та створення додаткових функціональних модулів	Таргетні пропозиції бізнесу, проведення презентації функціональних рішень на ярмарках та конференціях	Відсутність аналогічних до новостворених функціональних модулів у конкурентів	Розробка та удосконалення існуючих модулів на основі потреб ринку та інформації від клієнтів

В таблиці 5.16 наведено визначення стратегії позиціонування.

Таблиця 5.16 - Визначення стратегії позиціонування

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Можливі обидва варіанти	Стандартні функціональні модулі будуть виконувати схожі функції, що і конкуренти	Унікальна цінова політика, функціональні інновації

В таблиці 5.17 наведено визначення стратегії позиціонування.

Таблиця 5.17 - Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Висока якість прогнозування в клієнтській сфері застосування	Розробка та удосконалення існуючих модулів на основі потреб ринку та інформації від клієнтів	Спеціалізовані рішення, хмарні сервіси	Прогнозування, передбачення, аналіз

5.5 Розроблення маркетингової програми стартап-проекту

В таблиці 5.18 представлені ключові переваги концепції потенційного товару.

Таблиця 5.18 - Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Широкий функціонал	Вирішення задач	Забезпечує вирішення більшої кількості задач бізнесу
2	Спеціалізовані рішення	Вирішення задач	Забезпечує більш ефективно вирішення задач у звуженій сфері застосування
3	Технічні ресурси	Хмарні сервіси	Дозволяє користуватись рішенням за рахунок віддалених технічних потужностей

Опис трьох рівнів моделі товару відображено у таблиці 5.19.

Таблиця 5.19 - Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Обробка, аналіз даних. Прогнозування та передбачення потреб споживача

Продовження таблиці 5.19

Рівні товару	Сутність та складові		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	Швидкодія	Нм	Тх/Тл/Е
	Ефективність	Нм	Тх/Тл
	Користувацький інтерфейс	Нм	Е
	Якість: стандарти відповідні до законодавства. Створені функціональні скріпти.		
	Пакування: Власний сайт		
	Марка: RFS solutions, RFS		
III. Товар із підкріпленням	До продажу: застосунок для інтеграції в існуючі системи керування підприємством для прогнозування та передбачення потреб споживачів на основі великих масивів даних		
	Після продажу: Швидкодія, ефективність, легкість у користуванні		
Закритий код. Захищений від можливості декомпіляції.			

Визначення меж встановлення ціни показано в таблиці 5.20.

Таблиця 5.20 - Визначення меж встановлення ціни

№ п/п	Рівень цін на товари- замінники	Рівень цін на товари- аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	-	200\$/міс	Рівень доходів підприємств надзвичайно високий	150-200\$/міс

Формування системи збуту зображено в таблиці 5.21.

Таблиця 5.21 - Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптим альна система збуту
1	Таргетні пропозиції для компаній	Презентації функціоналу	-	-

Концепція маркетингових комунікацій відображена у таблиці 5.22.

Таблиця 5.22 - Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонува ння	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Середній бізнес – оптимальні рішення за невисоку ціну	Соціальні мережі, внутрішньо ринкова комунікація	Прогнозува ння покупок споживача	Короткий опис переваг продукту, заохочення дізнатись більше	Передбачення покупок споживачів

Продовження таблиці 5.22

№ п/п	Специфіка поведінки цілових клієнтів	Канали комунікацій, якими користуються цілові клієнти	Ключові позиції, обрані для позиціонува ння	Завдання рекламного повідомлення	Концепція рекламного звернення
2	Великий бізнес – повноцінні рішення для покращення продажів	Таргетні дзвінки до клієнтів	Прогнозува ння покупок споживача	Донести інформацію про оптимальність рішення для бізнесу клієнта	Передбачення покупок споживачів

Висновки до розділу

Отже, відповідно до вищенаведених результатів, можна стверджувати про наявність попиту на запропоновану систему. Варто зауважити, що на ринку присутня сильна конкуренція, але інноваційна складова продукту дозволяє суттєво збільшити конкурентоспроможність проекту. Подальша імплементація проекту доцільна за умови фокусування на середній та великий бізнес, оскільки саме цей ринок динамічно зростає і є достатньо рентабельним.

ВИСНОВКИ

В ході дослідження прояснено передумови виникнення кризових явищ, визначено поняття кризи, здійснений огляд основних підходів та методологій виявлення кризових явищ.

Побудовано прихований Марківський ланцюг для виявлення кризових явищ. Прослідковано еволюцію моделі від звичайного марківського ланцюга – до прихованого ланцюга Маркова в неперервному часі з Гаусівськими компонентами. З застосуванням динамічного програмування було отримано швидкі формули для тренування моделі, а також для здійснення прогнозів. На основі ЕМ-алгоритму було отримано модифікації цих формул для прихованого ланцюга Маркова в неперервному часі.

В майбутніх дослідженнях можливе подальше вдосконалення побудованої моделі. Перший напрямок до вдосконалення – при створенні моделі з кількома компонентами розглянути інші розподіли (у тому числі дискретні) для вирішення конкретної задачі. Інший напрямок – накласти на функцію правдоподібності додаткову вагову функцію залежності від часу, аби останні спостереження мали більшу вагу – це дасть змогу обійти можливі недоліки врахування Марківського припущення.

ПЕРЕЛІК ПОСИЛАНЬ

1. Markov Chains In Predictive Models of Currency Crises - With Applications to Southeast Asia / [R. S. Mariano, A. G. Abiad, B. Gultekin and el.]. – Singapore: SMU Economics & Statistics Working Paper Series, 2002. – 35 p.
2. Hidden Markov Model of Portfolio Credit Risk / [S. Ahuja, S. Bunsupha, K. Tan and el.]. – New Jersey: Englewood Cliffs, 2011. – 17 p.
3. Bilmes J. A. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models / Jeff Bilmes. – Ottawa: Advanced Symbolics, 1998. – 13 p.
4. Wagner. C. S. Presidential Election Forecasts: Through the Lense of Linear Algebra [Електронний ресурс] / Cassia S. Wagner // Electronic Thesis and Dissertation Repository. – 2012. – Режим доступу до ресурсу: <http://docplayer.net/26233199-U-s-presidential-election-forecasts-through-the-lense-of-linear-algebra-cassia-s-wagner.html>.
5. Constant A. The Dynamics of Repeat Migration: A Markov Chain Analysis / A. Constant, K. . Zimmermann. – Bonn: CEPR, 2004. – 39p.
6. Eisner J. An interactive spreadsheet for teaching the forward-backward algorithm [Електронний ресурс] / Jason Eisner // Department of Computer Science Johns Hopkins University. – 2002. – Режим доступу до ресурсу: <https://cs.jhu.edu/~jason/papers/eisner.tnlp02.pdf>.
7. McCallum A. Hidden Markov models Baum Welch algorithm [Електронний ресурс] / Andrew McCallum // College of Information and Computer Sciences. – 2004. – Режим доступу до ресурсу: <https://people.cs.umass.edu/~mccallum/courses/inlp2004a/lect10-hmm2.pdf>.
8. Ramage D. Hidden Markov Models Fundamentals [Електронний ресурс] / Daniel Ramage// College of Information and Computer Sciences. – 2007. –

Режим доступа до ресурсу:
<https://people.cs.umass.edu/~mccallum/courses/inlp2004a/lect10-hmm2.pdf>.

9. Schrod P. A. Forecasting Conflict in the Balkans using Hidden Markov Models [Электронный ресурс] / Philip Schrod P. A. // Parus Analytical Systems. – 2000. – Режим доступа до ресурсу:
<http://parusanalytics.com/eventdata/papers.dir/forecasting.html>

10. Schrod P. A. Pattern Recognition of International Crises using Hidden Markov Models [Электронный ресурс] / Philip Schrod P. A. // Parus Analytical Systems.. – 1997. – Режим доступа до ресурсу:
<http://eventdata.parusanalytics.com/papers.dir/schro97b.pdf>

11. Schrod P. A. A Early Warning of Conflict in Southern Lebanon using Hidden Markov Models [Электронный ресурс] / Philip Schrod P. A. // Parus Analytical Systems. – 1999. – Режим доступа до ресурсу:
<http://eventdata.parusanalytics.com/papers.dir/schro99b.pdf>

12. Sweatha B. Using smoothing techniques to improve the performance of Hidden Markov's Model [Электронный ресурс] / Boodidhi Sweatha // UNLV Libraries. – 2011. – Режим доступа до ресурсу:
<http://digitalscholarship.unlv.edu/cgi/viewcontent.cgi?article=2008&context=thesesdissertations>.

13. Tenyakov A. Estimation of Hidden Markov Models and Their Applications in Finance [Электронный ресурс] / Anton Tenyakov // Electronic Thesis and Dissertation Repository. – 2014. – Режим доступа до ресурсу:
<http://ir.lib.uwo.ca/etd/2348/>.

14. Zhang Y. J. Prediction of financial time series with hidden Markov models [Электронный ресурс] / Ying jian Zhang // Simon Fraser University. – 2001. – Режим доступа до ресурсу:
https://www.cs.sfu.ca/~anoop/students/rzhang/rzhang_msc_thesis.pdf.

15. Гурьянова Л. С. Модели прогнозирования резонансного взаимодействия экономических индикаторов / Л. С. Гурьянова // Проблемы і перспективи розвитку банківської системи – №6. – С. 59–65.

16. Поліщук О. В. Сутність поняття валютної кризи та її вплив на Україну / О. В. Поліщук // Економічні перспективи України та світу. – 2013. – №3. – С. 22–23.

17. Проноза В. Моделювання розвитку кризових процесів в реальному секторі економіки України / В. Проноза. // Регіональна економіка. – 2014. – №1. – С. 7–15.

18. Пустовіт Д. Т. Марківський підхід в задачі передбачення кризових явищ ринку праці в Україні // Міжнародний науковий журнал "Інтернаука". — 2017. — №10. – С. 81–95.

19. Яценко Г. Моделювання трансмісії кризових явищ в економіці України. Прикладна економіка / Ганна Яценко // Економіст. – 2014. – №3. – С. 51–56.

ДОДАТОК А ЛІСТИНГ ПРОГРАМИ

```

var Mathjs = require('./math.js');
var math = new Mathjs();

module.exports = function (hmmModel, observations, beta) {
    var row, column, time, nextLineColumn, sum;
    var N = hmmModel.getStatesSize();
    var T = observations.length;
    var emmisionProbability = hmmModel.getEmissionProbability();
    var transitionProbability = hmmModel.getTransitionProbability();

    for (row = 0; row < T; row++) {
        beta[row] = [];
    }

    // 1. Initialization
    for (column = 0; column < N; column++) {
        beta[T-1][column] = 1;
    }

    // 2. Induction
    for (time = T - 2; time >=0; time--) {
        for (column = 0; column < N; column++) {
            sum = 0;
            for (nextLineColumn = 0; nextLineColumn < N;
nextLineColumn++) {
                sum += transitionProbability[column][nextLineColumn]
*
                emmisionProbability[nextLineColumn][observations[time+1]] *
                beta[time+1][nextLineColumn];
            }
            beta[time][column] = sum;
        }
    }
}

```

```

// 3. Termination
var result = 0;
for (column = 0; column < N; column++) {
    result += beta[1][column];
}
return math.log2(result);
};

var forward = require('./forward');
var backward = require('./backward');

var DELTA = 0.1;
// Baum-Welch Algorithm
module.exports = function (hmmModel, observations) {

    // 1. initialization
    var pi = [];
    var alpha = [];
    var beta = [];
    var transitionProbability = [];
    var emissionProbability = [];

    var N = hmmModel.getStatesSize();
    var M = hmmModel.getObservationsSize();
    var T = observations.length;
    var row, column, time, denominatorA, denominatorB, numeratorA,
    numeratorB, logProbPrev, delta;
    var initPi = 1 / N;

    // 1.1 init pi
    for (column = 0; column < N; column++) {
        pi[column] = initPi;
        transitionProbability[column] = [];
        emissionProbability[column] = [];
        alpha[column] = [];
        beta[column] = [];
    }

    // 1.2 init alpha
    var logprob = forward(hmmModel, observations, alpha);

```

```

// 1.3 init beta
var logprobb = backward(hmmModel, observations, beta);

var gamma = computeGamma(T, N, alpha, beta);
var xi = computeXi(hmmModel, observations, alpha, beta);

logProbPrev = logprobb;
delta = 1;

// 2. EM training induction until delta < DELTA
// do {
while (delta > DELTA) {
    // 2.1 compute pi
    for (column = 0; column < N; column++) {
        pi[column] = 0.001 + 0.999*gamma[1][column];
    }

    // 2.2 compute transition matrix and emission matrix
    for (row = 0; row < N; row++) {
        denominatorA = 0;
        for (time = 0; time < T-2; time++)
            denominatorA += gamma[time][row];

        for (column = 0; column < N; column++) {
            numeratorA = 0;
            for (time = 0; time < T-2; time++)
                numeratorA += xi[time][row][column]
            transitionProbability[row][column] = 0.001 +
0.999*numeratorA/denominatorA;
        }

        denominatorB = denominatorA + gamma[T-1][row];
        for (var k = 0; k < M; k++) {
            numeratorB = 0;
            for (time = 0; time < T; time++) {
                if (observations[time] == k)
                    numeratorB += gamma[time][row];
            }
            emissionProbability[row][k] = 0.001 + 0.999 *
numeratorB / denominatorB;

```

```

        }
    }

    hmmModel.setStartProbability(pi);
    hmmModel.setTransitionProbability(transitionProbability);
    hmmModel.setEmissionProbability(emissionProbability);

    // 2.3 comput logfile
    logprobf = forward(hmmModel, observations, alpha);
    logprobb = backward(hmmModel, observations, beta);
    gamma = computeGamma(T, N, alpha, beta);
    xi = computeXi(hmmModel, observations, alpha, beta);

    delta = Math.abs(logprobf - logProbPrev);
    logProbPrev = logprobf;

    } // while (delta > DELTA);
    return hmmModel;
};

// compute  $\gamma$ 
// T is number of observation
// N is number of states
function computeGamma (T, N, alpha, beta) {
    var column, row, time, denominator;
    var gamma = [];
    for (time = 0; time < T; time++) {
        gamma[time] = [];

        for (time = 0; time < T; time++) {
            denominator = 0;
            for (column = 0; column < N; column++) {
                gamma[time][column] = alpha[time][column] *
beta[time][column];

                denominator += gamma[time][column];
            }

            for (column = 0; column < N; column++) {
                gamma[time][column] = gamma[time][column] / denominator;
            }
        }
    }
}

```

```

        }
    }
    return gamma;
};

// compute  $\xi$ 
function computeXi (hmmModel, observation, alpha, beta) {
    var xi = [];
    var T = observation.length;
    var N = hmmModel.getStatesSize();
    var transitionProbability = hmmModel.getTransitionProbability();
    var emissionProbability = hmmModel.getEmissionProbability();
    var columnI, columnJ, time, sum;
    for (time = 0; time < T; time++) {
        xi[time] = [];
        for (columnI = 0; columnI < N; columnI++) {
            xi[time][columnI] = [];
            for(columnJ = 0; columnJ < N; columnJ++) {
                xi[time][columnI][columnJ] = [];
            }
        }
    }

    for (time = 0; time < T - 1; time++) {
        sum = 0;
        for (columnI = 0; columnI < N; columnI++) {
            for (columnJ = 0; columnJ < N; columnJ++) {
                xi[time][columnI][columnJ] = alpha[time][columnI]
                    * beta[time+1][columnJ]
                    * transitionProbability[columnI][columnJ]
                    *
emissionProbability[columnJ][observation[time+1]];
                sum += xi[time][columnI][columnJ];
            }
        }

        for (columnI = 0; columnI < N; columnI++) {
            for (columnJ = 0; columnJ < N; columnJ++) {
                xi[time][columnI][columnJ] =
xi[time][columnI][columnJ] / sum;
            }
        }
    }
}

```

```

        }
    }
}

return xi;
};

var Mathjs = require('./math.js');
var math = new Mathjs();

module.exports = function (hmmModel, observations, alpha) {
    var row, column, time, prevColumn, sum;
    var N = hmmModel.getStatesSize();
    var T = observations.length;
    var pi = hmmModel.getStartProbability();
    var emissionProbability = hmmModel.getEmissionProbability();
    var transitionProbability = hmmModel.getTransitionProbability();

    for (row = 0; row < T; row++) {
        alpha[row] = [];
    }

    // 1. Initialization
    for (column = 0; column < N; column++) {
        alpha[0][column] =
pi[column]*emissionProbability[column][observations[0]];
    }

    // 2. Induction
    for (time = 1; time < T; time++) {
        for (column = 0; column < N; column++) {
            sum = 0;
            for (prevColumn = 0; prevColumn < N; prevColumn++) {
                sum += alpha[time-
1][prevColumn]*transitionProbability[prevColumn][column];
            }
            alpha[time][column] =
sum*emissionProbability[column][observations[time]];
        }
    }
}

```

```

    }

    // 3. Termination
    var result = 0;
    for (column = 0; column < N; column++) {
        result += alpha[T-1][column];
    }
    return math.log2(result);
};

module.exports = Mathjs;

function Mathjs () {

};

Mathjs.prototype.log2 = function (val) {
    return Math.log(val) / Math.LN2;
};

module.exports = Model;

function Model () {
    this.statesSize = null;
    this.observationsSize = null;
    this.startProbability = null;
    this.transitionProbability = null;
    this.emissionProbability = null;
};

Model.prototype.setStatesSize = function (statesSize) {
    this.statesSize = statesSize;
};

Model.prototype.getStatesSize = function () {
    return this.statesSize;
};

Model.prototype.setObservationsSize = function (observationsSize) {
    this.observationsSize = observationsSize;
};

```

```

};

Model.prototype.getObservationsSize = function () {
    return this.observationsSize;
};

// startProbability: [0.6, 0.4]
Model.prototype.setStartProbability = function (startProbability) {
    this.startProbability = startProbability;
};

Model.prototype.getStartProbability = function () {
    return this.startProbability;
};

// transitionProbability : [
//     [0.7, 0.3],
//     [0.4, 0.6]
// ]
Model.prototype.setTransitionProbability = function (transitionProbability) {
    this.transitionProbability = transitionProbability;
};

Model.prototype.getTransitionProbability = function () {
    return this.transitionProbability;
};

// emissionProbability : [
//     [0.1, 0.4, 0.5],
//     [0.6, 0.3, 0.1]
// ]
Model.prototype.setEmissionProbability = function (emissionProbability) {
    this.emissionProbability = emissionProbability;
};

Model.prototype.getEmissionProbability = function () {
    return this.emissionProbability;
};

// calculate every node delta(i, t)
// then choose maxDelta(t)

```



```

// trace back to t=1, all value in path is the viterbi result
// observations is a token array
module.exports = function (hmmModel, observations) {
    var delta = []; // value
    var psi = []; // psi  $\psi$  store path
    var result = [];
    var T = observations.length; // number of observations
    var N = hmmModel.getStatesSize(); // number of states
    var pi = hmmModel.getStartProbability();
    var emissionProbability = hmmModel.getEmissionProbability();
    var transitionProbability = hmmModel.getTransitionProbability();
    var row, column, prevRowColumn, time, maxVal, maxValPath, val;

    /* 1. Initialization */
    for (row = 0; row < T; row++) {
        psi[row] = [];
        delta[row] = [];
        result[row] = -1;
    }

    for (column = 0; column < N; column++) {
        delta[0][column] = pi[column] *
emissionProbability[column][observations[0]];
        psi[0][column] = -1;
    }

    /* 2. Recursion */
    for (time = 1; time < T; time++) {
        for (column = 0; column < N; column++) {
            maxVal = 0;
            maxValPath = 0;
            for (prevRowColumn = 0; prevRowColumn < N; prevRowColumn++)
            {
                val = delta[time-1][prevRowColumn] *
transitionProbability[prevRowColumn][column];
                if (val > maxVal) {
                    maxVal = val;
                    maxValPath = prevRowColumn;
                }
            }
        }
    }
}

```

```

        delta[time][column] = maxVal *
emissionProbability[column][observations[time]];
        psi[time][column] = maxValPath;
    }
}

/* 3. Termination */
var probability = 0;
for (column = 0; column < N; column++) {
    if (probability < delta[T-1][column]) {
        probability = delta[T-1][column];
        result[T-1] = column;
    }
}

/* 4. psi (state sequence backtracking) */
for (time = T-2; time >= 0; time--) {
    result[time] = psi[time+1][result[time+1]];
}

return result;
}

```